TU MICRO

COMMODORE

N.º 8 · SEGUNDA EPOCA

500 PTAS · (IVA INCLUIDO)

PERFIL DEL AMIGA

TRES DIMENSIONES EN NUESTRO COMMODORE

CASA DE SOFTWARE PARA EL PROFESIONAL

EXTRA NAVIDAD

MAS PAGINAS

MAS CONTENIDO





TARDAREMOS EN VER ALGO PARECIDO.

En informática ya es difícil sorprender. Pues bien, Commodore lo ha conseguido con Amiga.

Más que una nueva generación de ordenadores, el Amiga de Commodore representa un nuevo concepto. Un ordenador que ofrece una serie de posibilidades y abre unos caminos que hasta hoy eran impensables.

Entre las novedades de este ordenador profesional destacan: la posibilidad de trabajar con un procesador de 32 bits y de 3 coprocesadores específicos, actuando los 4 al tiempo.

Además, Amiga dispone de pantallas y ventanas configurables con una resolución de 640 x 400 puntos escogiendo entre una paleta de 4.096 colores.

Trabaja con 4 canales independientes polifónicos con voz masculina y femenina, y por si fuera poco, Amiga de Commodore es el único ordenador multitarea que puede efectuar varios trabajos simultáneamente.

El precio también es importante. Por 330.000 ptas. ex. IVA, el Amiga incluye:

- Unidad central con 768 K RAM.
- Monitor de alta resolución en color y sonido.
- Unidad de discos de 880 K.
- Teclado profesional y ratón.

Por mucho que avance la investigación en el campo de los ordenadores, tardaremos en ver algo parecido.

commodore

Imagina siempre lo mejor.

© Bassat, Ogilvy & Mather

Microelectrónica y Control, S.A. *Unico representante de Commodore en España.*
c/ Valencia, 49-53 Tel.(93) 325 57 02 08015 Barcelona - c/ Ardemans, 8 Tel. (91) 246 37 36 28028 Madrid

Me puedes regalar...

una raqueta, un ajedrez,
un jersey, una estilográfica,
o una radio para el coche...

pero lo que yo necesito...
es este Commodore 128.



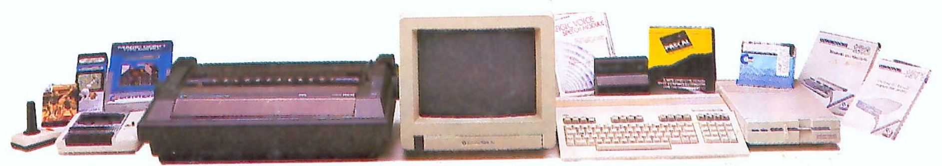
Y lo necesito porque es el más 128 de todo el mercado. Porque es 3 ordenadores en uno solo, sólo con pulsar una tecla.

Lo necesito porque es compatible con el C-64, y todos sus programas.

Lo necesito, porque ya que me decido, quiero lo mejor.

Lo necesito, porque todavía hay un par de vídeo juegos en los que me gana mi vecino y eso no puede ser.

commodore



Microelectrónica y Control, s. a. *Unico representante de Commodore en España. c/ Valencia, 49-53 08015 Barcelona - c/ Ardemans, 8 28028 Madrid*

Director:
ANTONIO M. FERRER ABELLO
Redactor Jefe:
FERNANDO LOPEZ MARTINEZ
Redacción:
ANTONIO CARVAJAL
JOSE LUIS DE DIEGO
JUAN M. LOPEZ MARTINEZ
PABLO GARCIA MOLINA
IGNACIO BARCO LUENGO
ALFREDO SINDIN VALERO
FERNANDO ACERO MARTIN
Colaboradores:
JOSE LUIS M. VAZQUEZ DE PARGA
ANTONIO MANZANERA
Secretaria de Redacción:
PILAR MANZANERA AMARO
Maquetación:
CARLOS GONZALEZ AMEZUA
CARLOS TALLANTE
Ilustraciones:
ANTONIO PERERA
RAMON POLO
Fotografía:
EQUIPO GALATA
Directora Publicidad:
CARMINA FERRER
Publicidad Madrid:
BEGOÑA LLORENTE
Tel.: (91) 457 69 23
Publicidad Barcelona:
ISIDRO IGLESIAS
Avda. Corts Catalanes, 1010
Tel.: (93) 307 11 13
Director de Producción:
VICENTE ROBLES
Directora de Administración:
MARIA ANTONIA BUITRAGO
Suscripciones:
MARIA GONZALEZ AMEZUA
Redacción, administración, publicidad y suscripciones:
Plza. República Ecuador, 2.
28016 MADRID. Tel.: 457 94 24
Télex 49371 ELOC E
Dirección para correspondencia:
Aptdo. de Correos 61.294
28080 MADRID
TU MICRO COMMODORE es una publicación mensual de Ediciones INGELEK.

Fotomecánica:
RODACOLOR, S. A.
Impresión:
GRAFICAS REUNIDAS, S. A.
Distribución:
COEDIS, S. A.
Valencia, 245. Tel.: 215 70 97
08007 BARCELONA
Almacén: Nacional II. Km. 609,4
MOLINS DE REI (Barcelona)
Delegación en Madrid: Serrano, 165.
Tel.: 411 11 48
Almacén: Laforja, 19-21, esq. Hierro
Pol. Industrial Loeches
TORREJON DE ARDOZ (Madrid)

Precios para España. Ejemplar: 500 ptas. (IVA incluido). 485 ptas. (Canarias, Ceuta y Melilla).
(La suscripción anual incluye 11 números).
Distribución Cono Sur:
CADE, S.R.L.
Pasaje Sud América, 1532
Tel.: 21 24 64
Buenos Aires 1.290. Argentina.

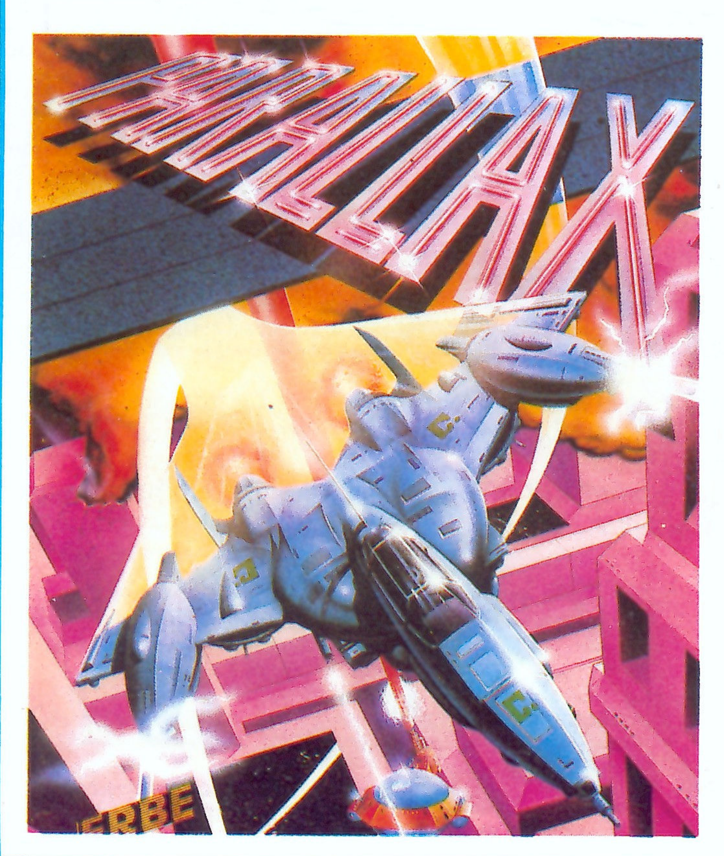
Depósito Legal: M. 40920-1985.
Impreso en España

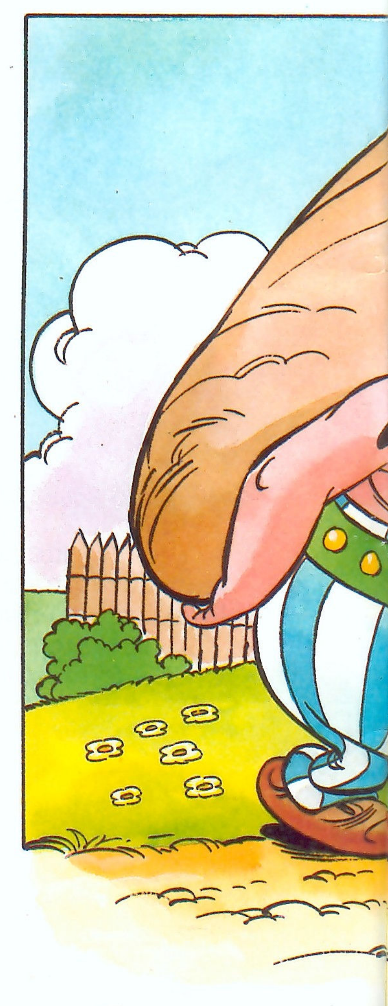
diciembre 1986

Para manejar la unidad de disco y los datos almacenados en código máquina es indispensable conocer el BASIC y el lenguaje ensamblador del microprocesador 6510. Este interesante artículo nos apoyará en esta tarea, con transcripciones al lenguaje BASIC de las rutinas de demostración que lo acompañan.



Lo que parecía un mundo amigable en una primera apreciación, se ha convertido de repente en una ratonera de la que te será muy difícil salir. Ya no se puede echar marcha atrás, el planeta se revela como es verdaderamente y Parallax muestra sus garras.

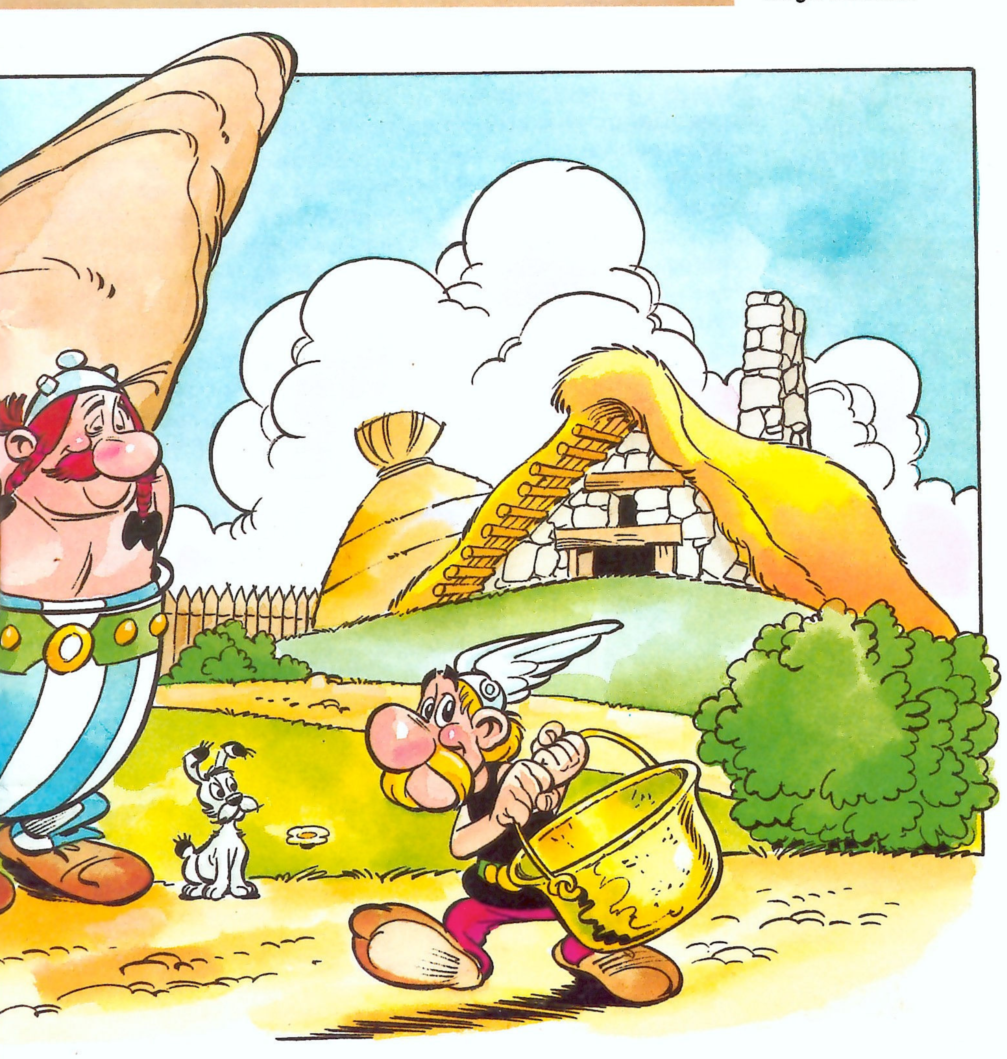






Destinado a ocupar el vacío existente en Commodore, en el segmento medio-alto, el Amiga ha dado pie a ríos de tinta en su país de origen: U.S.A. Estudiemos sus peculiaridades.

Este mes tenemos el honor de dedicar unas páginas al famosísimo guerrero Asterix. La razón es muy simple: Melbourne House ha lanzado al mercado un entretenido juego en el que aparece como protagonista el citado galo: Asterix and the magic cauldron.



SUMARIO

La Navidad es siempre una época especialmente propicia para que todos nos sintamos con ánimo de resolver los problemas de los demás... o al menos intentarlo. Ha llegado a nuestra redacción, desde Argentina, una carta muy peculiar; su contenido: la angustia de un joven lector cuyo único problema es la marginación de sus compañeros. ¿Por qué?, simplemente porque él no tiene un Commodore. Así que, ¿a ninguno de nosotros nos sobra un Commodore?, ¿nadie se ha cambiado de equipo porque el antiguo se le ha quedado pequeño? Miremos por los rincones a ver si encontramos alguno entre todos. ¡Feliz Navidad!

noticias

Idealogic

Idealogic presentaba novedades, tanto en hardware como en software. En el campo del hardware destacaron las tarjetas de reles y de introducción Omnibus, ambas destinadas al ámbito de la formación profesional, especialistas y hobbystas.

En lo que se refiere al software, presentaba los programas de Dimension New, Invierte y Gana, y Risky Holding, así como los educativos de la serie Pitágoras de S.M.



Enhorabuena a los ganadores

Este mes los agraciados ganadores de una suscripción por un año a nuestra revista TU MICRO COMMODORE, por la ayuda prestada en la confección de la sección LOS 7 MAGNIFICOS, han sido los siguientes lectores:

Juan Gabriel Castillo Román, Barcelona.

Francisco Avila Parra, Málaga.

Juan Ferrer Abad, Zaragoza.

José Miguel de la Mata Vicente, Alcalá de Henares (Madrid).

Gustavo A. Boyero, Salamanca.

La próxima ocasión puede llegarte el turno a ti; ¡anímate y mándanos tu voto!

Enfa Ibérica

Aprovechamos la ocasión para aclarar a nuestros lectores un pequeño mal entendido surgido a raíz de la publicación del artículo TUS MICRO ACCESORIOS, en el pasado mes de noviembre. Tal y como anunciábamos en el mismo Enfa Ibérica DISTRIBUYE a unos precios ciertamente razonables una serie de accesorios de gran utilidad.

No obstante, esta firma no tiene departamento de venta al público, sino que la comercialización de sus productos se lleva a cabo a través de la red de tiendas. Lógicamente, los comerciantes también tienen derecho a su margen de beneficio ¡razonable! ¿no? y adicionalmente, en el trasiego distribuidor-comerciante siempre surgen los inevitables gastos; resultado: el precio de venta al público difiere, lógicamente, del de distribución que publicamos en nuestra revista, y como esta redacción considera más útil conocer el primer dato que el segundo, ahí va una lista de PVP aproximados, que sin duda nos hará falta conocer para las compras de estas navidades:

Cubreteclados 1: 2.300
Cubreteclados 2: 2.200
Archivador discos: 4.400
Clean check: 700
Tacko: 1.300
Soporte para monitor: 4.400

Que nadie se llame a engaño: todos estos precios son aproximados, y además habrá que sumarles el IVA correspondiente. En todo caso una ganga, si tenemos en cuenta que por unas 1.500 ptas. podremos duplicar nuestra discoteca, etc., ahora eso sí, de la ganga a lo ilusorio ¡va una diferencia!

Micro Electrónica y control

La atención de los commodorianos que visitaban el stand de MEC se centraba, como viene siendo habitual de un tiempo a esta parte, en el siempre impresionante Amiga. No nos cansaremos de repetir que ver un Amiga funcionando con un buen programa de demostración, como aquellos de los que hace gala Micro Electrónica, es una experiencia inigualable que realmente merece la pena tener.

Por lo demás, no había grandes novedades, aparte de la presencia en el stand del nuevo Commodore 64 C con su «new look» C-128, así como la unidad de disco 1541, que también había recibido un baño de color crema.

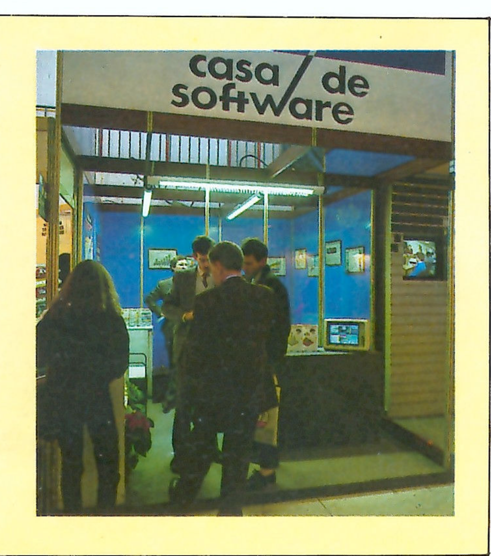
Para no apartarse de la tónica PC reinante en la feria (la verdad es que en vez del SIMO, parecía un aquarium: (¡todo lleno de PeCes!), hacía acto de presencia la tarjeta Sidecard, de emulación por software de un PC para Amiga, al precio de 139.000 ptas.

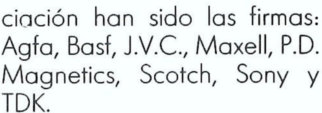
En todo caso, quizá lo más interesante fue la entrevista que mantuvimos con D. Josep Riera, de Micro Electrónica y Control, que por motivos de espacio publicaremos en nuestro próximo número.

Simo 86

La feria SIMO 86, celebrada en Madrid del 14 al 21 de noviembre, en el recinto ferial de la Casa de Campo Ifema, contó con un gran número de expositores que ocuparon seis pabellones. Haremos a continuación un resumen de las novedades más importantes que para el mercado de los usuarios Commodores se presentaron en la exposición y con motivo de la misma, que por otra parte, fueron en esta edición algo escasas.







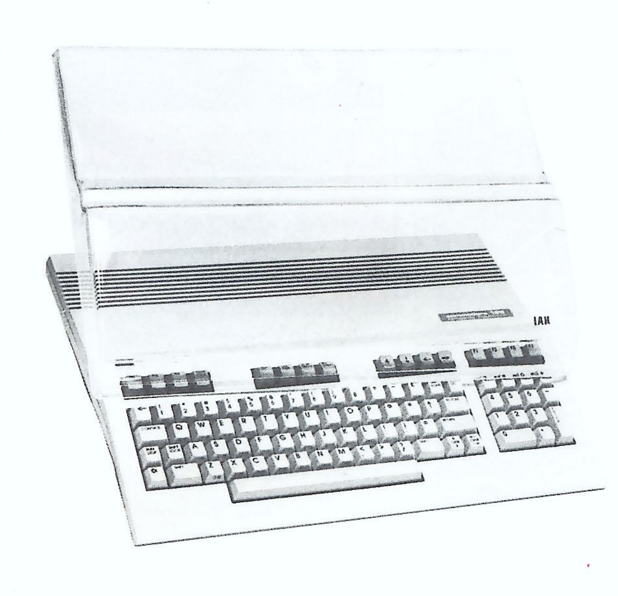


Más productos de la firma First

El nuevo software COPY II 64 de Central Point Software representada por First, para Commodore 64, realiza la función de copias de seguridad del software del usuario o «back up».

Precisa de un ordenador Commodore 64 y una unidad de disco 1541, permitiendo hacer copias de seguridad de programas protegidos, en menos de 3 1/2 minutos (tanto en discos protegidos o no), con el máximo automatismo, sin necesidad de tener conocimientos técnicos, ya que se incluye una lista completa de software, con los parámetros correspondientes.

Se dispone de las siguientes opciones: copia de disco entero, con indicación de inicio y final de copia de pistas o medias pistas, e incremento medio o entero de éstas y modificación de parámetros.



El precio de Copy II es de 9.890 e incluye un diskette de 5 1/4" y manual.

Otro producto también de la firma First, esta vez para el Commodore 128, es un cubreteclado: Ian Key Cover, protector ideal para tu ordenador. Realizado con resistente metacrilato de una pieza, que resguardará al ordenador del polvo, derrame de líquidos, golpes, humedad excesiva, etc., manteniéndolo siempre limpio. Su precio es de 2.576 ptas. Lo puedes, encontrar al igual que el Copy II, en la calle Valencia, 195 de Barcelona. Teléfono: (93) 323 03 90.

ANFIMA: una nueva asociación

Con motivo de la celebración de Sonimag, tuvo lugar la primera Asamblea General de la Asociación Empresarial Española de Fabricantes e Importadores Oficiales y/o Exclusivos de Soportes Magnéticos Virgenes (cintas audio, vídeo e informática), ANFIMA

Anfima fue constituida como Asociación en el pasado mes de julio y ha obtenido las autorizaciones legales pertinentes para su adecuado funcionamiento. Tiene su domicilio social en Barcelona, en el número 71 Bis de la calle Bailen.

Los promotores de la Asociación han sido las firmas: Agfa, Basf, J.V.C., Maxell, P.D. Magnetics, Scotch, Sony y TDK.

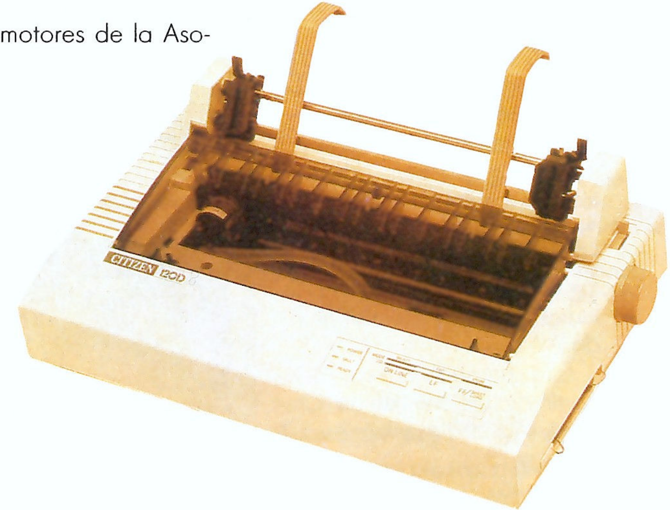
Su objetivo y fin consiste en actuar ante la Administración Estatal y Autonómica, en cuantas cuestiones afecten directamente a las cintas de audio, vídeo e informática, llenando el vacío existente en este campo, siguiendo el ejemplo de las Asociaciones en funcionamiento tanto en la C.E.E., como en el resto del mundo.

Delegación Tesin en Madrid

La firma TESIN, S.A. ha abierto recientemente una delegación en Madrid, en la calle Comandante Zorita, 13, oficina 114 con los teléfonos (91) 254 19 25 y 254 16 83.

Su sede central se encuentra en la calle Provenza, 10-12. Barcelona 08029 y sus teléfonos son (93) 322 44 61 y 321 67 12 y 321 61 53.

Tesin es el importador oficial de las impresoras Citizen, que posee un modelo especial para Commodore, el 120 DC, que imprime en 80 columnas con una velocidad de 120 caracteres por segundo (cps), en calidad normal y 25 cps en alta calidad, con 4K de buffer, al precio de 65.975 sin IVA.



NOTICIAS

Nueva tienda Konami

Ha sido abierta al público la nueva tienda Konami, sita en la calle Francisco de Navacerrada, 19 de Madrid.

Tiene la peculiaridad de estar dividida por sistemas, con stands especiales para tu ordenador. Allí también podrás probar los juegos en máquinas Arcade (como las de los recreativos), pero gratis.



Con la garantía de tu ordenador (no importa la fecha de compra), podrás obtener el carnet de socio de la tienda Konami, con el cual tendrás acceso a concursos, sorteos y descuentos. A los 3.000 primeros socios les será entregado un «Regalo Sorpresa» junto con el carnet de Konami Shop.

Convención en Francia de piratas informáticos

El periódico francés «Le Matin», después del contacto con un grupo de piratas informáticos o «hackers», demostró que cualquier consola de ordenador medio conectada con un modem, puede forzar las cadenas informáticas más importantes de todo el mundo.

Esta prueba se realizó a partir de los Minitel de Correos, y permitió entrar en 20 grandes ordenadores de todo el mundo, todo ello en presencia de tres expertos: un jurista, un especialista en informática y un experto en fraudes bancarios. Además, encontraron el medio de hacer pagar la factura telefónica de la conexión, a sus propias víctimas.

Los «hackers», cuyo decano no cuenta más de dieciocho años de edad, tardaron sólo una hora en penetrar en la máquina de una universidad belga, donde dejaron colocado un «espía», un pequeño programa que registra todas las comunicaciones del ordenador y sus respectivas palabras de paso. En caso de que éstas sean cambiadas, el «espía» registrará los nuevos códigos permitiendo nuevas intrusiones en el sistema.

Según el experto informático presente en la experiencia la mayor parte de las entradas en el ordenador no logran penetrar hasta las memorias internas del mismo, y lo único que consiguen es bloquearlo.

El jurista declaró después de esta convención, que todo lo que no está prohibido, está permitido. Esta máxima es el horror de los moralistas y el paraíso de los juristas y los perversos.

Convirtamos nuestro Commodore en un refinado muestreador

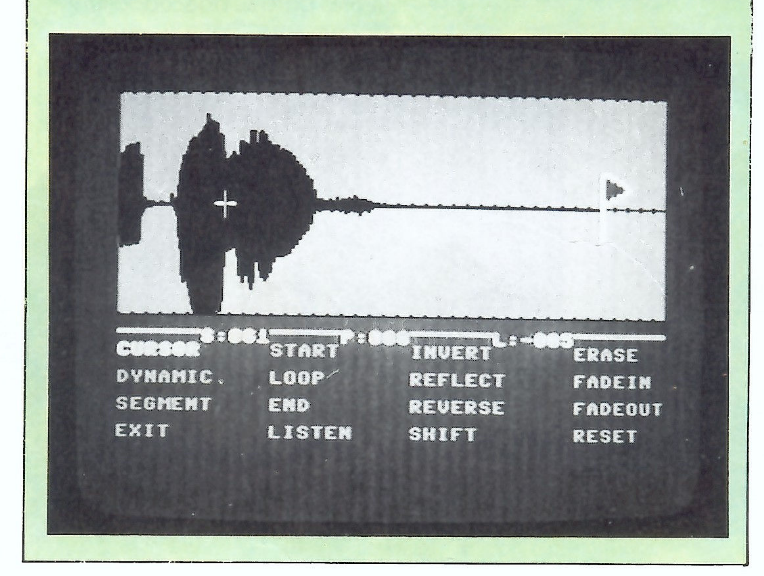
Ahora es posible digitalizar el chirrido de una puerta, el motor de un coche, el ladrido de un perro, o la voz de una persona saludando, tratando el sonido resultante como si de un instrumento musical se tratara.

Conectando nuestro Commodore 64 ó 128 a un Microvox, tomando muestras de cualquier sonido (8 velocidades de muestreo seleccionables con una banda de 20Khz y 0.9 segundos en la más veloz) y nos podremos asombrar de las posibilidades de este accesorio.

Podremos almacenar hasta 16 voces en la memoria, accediendo a ellas a través del secuenciador de 2.000 notas que se incluye en el software de la unidad, o utilizando los diskettes para el almacenamiento de nuestras muestras preferidas.

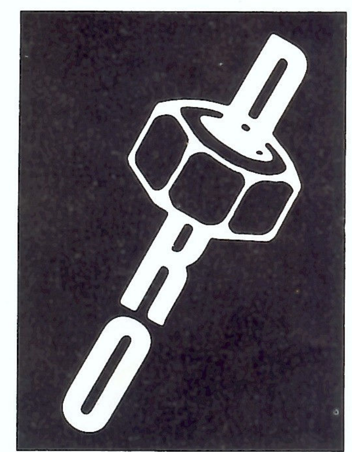
Microvox permite generar otros efectos especiales como armonización, flanger/phasing, eco y la toma de muestras múltiples o extremadamente largas de hasta 19.9 segundos. El sistema compander que incorpora le proporciona, a pesar de utilizar sólo 8 bits, 88 dB de relación señal/ruido y un rango dinámico de 72 dB.

El importador de este producto es Sistemas Midi, sito en la calle Córcega, 89 entlo. 08029 Barcelona con teléfono (93) 230 97 90. Su precio es de 75.000 ptas.



Noticias Software

Dro Soft ha firmado contrato en exclusiva para la fabricación y distribución en España, de los productos desarrollados por las firmas EDGE y VIRGIN, ambas británicas.



Asimismo, y siguiendo esta tónica de relaciones internacionales, Proeinsa ha llegado al acuerdo de distribución en exclusiva para España de los programas de la firma francesa Loriciels. De igual modo, es noticia la comercialización de nuevos títulos, que vendrán a engrosar el ya extenso catálogo de software Proeinsa: Hijack, Dandy, Prodigy, Basket, Aliens, Laberynth, Big trouble in little China y Howard the duck.

COMMODORE

¡NO HAY COMPETENCIA POSIBLE!

en Calidad / Precio







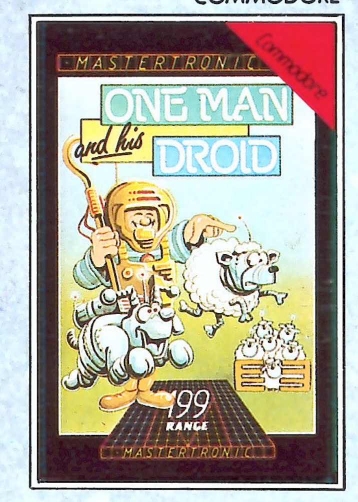
MSX
AMSTRAD
COMMODORE



SPEED KING

El juego de carreras de Motocicletas con la emocionante acción de correr rueda con rueda contra otros 19 pilotos. ¡Ponte el casco y vive la inolvidable aventura de las motos de altas prestaciones compitiendo a 250 millas a la hora!

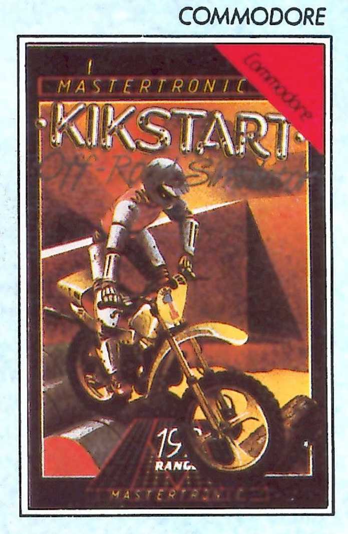
AMSTRAD
SPECTRUM
COMMODORE



ONE MAN AND HIS DROID

Viaja al planeta Andromadous con tu androide y captura a todos los ramboides que puedas. Hazlo rapidamente, tienes un tiempo limitado.

COMMODORE



KIKSTART

El simulador de las motos todo-terreno. Uno o dos jugadores. Selección de diferentes pistas.

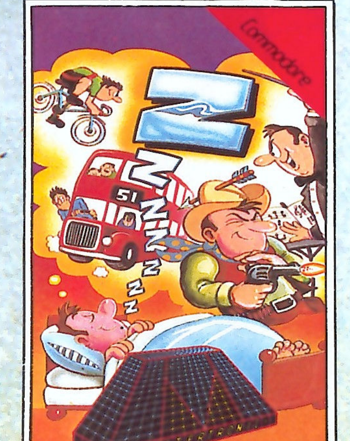
COMMODORE



NINJA

Arruina a todos los cinturones negros de los demás juegos de Artes Marciales.

SPECTRUM
COMMODORE



ZZZ

Estas atrapado en tu sueño y para escapar de esta aventura gráfica controlada por iconos y textos, tienes que encontrar la aduana o vagabundear por el mundo de ZZZ ¡para siempre, jamás!.

COMMODORE
AMSTRAD

FIVE-A-SIDE SOCCER

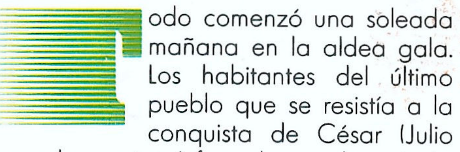
Fútbol Sala. Diez viciosos minutos de auténtico Fútbol. Con cinco jugadores por equipo, gráficos tridimensionales, uno o dos jugadores, increíble sintetizador de voz, lanzamiento de penalty, la más rápida animación ...



Licencia exclusiva para ESPAÑA DRO SOFT
Fundadores, 3 - 28028-MADRID
Tels. 255 45 00/09

Asterix y el Caldero Mágico

Este mes tenemos el honor de dedicar unas páginas al famosísimo guerrero Asterix (si hay alguien que no lo conozca, que levante la mano... Bien, continuemos). La razón es muy simple: Melbourne House ha lanzado al mercado un entretenido juego en el que aparece como protagonista el citado galo, héroe entre sus compatriotas y pesadilla continua del ¿terrible? ejército romano.



Todo comenzó una soleada mañana en la aldea gala. Los habitantes del último pueblo que se resistía a la conquista de César (Julio para los amigos) formaban cola ante la marmita humeante, removida sabiamente por el anciano druida Panoramix. Este último (a juicio de gran cantidad de historiadores) era el verdadero responsable del triunfo de aquel pequeño núcleo de galos, pues «preparaba una poción mágica que dotaba a quien la bebía de una fuerza sobre humana [...] aunque los efectos de ésta se desipaban al poco tiempo». (Herodotix, «Los druidas en la tradición gala»).

Pues bien, aquella parecía una mañana como otras tantas, en la que los valientes guerreros se disponían a tomar su ración de poción mágica para zurrar de lo lindo a los romanos que, temblando y sudando de miedo, esperaban en los campamentos fortificados el desarrollo de aquella ancestral costumbre. «Siempre es bueno comenzar el día con un poco de ejercicio», había dicho en más de una ocasión Abraracurcix, el jefe de la tribu.

Entre los que esperaban su turno se podía apreciar una obesa figura (¿gordo yo?, ¡que va!, sólo un poco entradito en carnes) seguida de un simpático perrito. Cuando llegó a la marmita, Panoramix le dijo:

— ¡No Obelix! (los avispados lectores que hayan adivinado la identidad del personaje, amigo inseparable de Asterix y repartidor de menhires de oficio, antes de leer este párrafo, que envíen sus datos, talla de calzado y árbol genealógico completo a esta editorial, ya que sortearemos entre los 125.000 primeros, 400 carnets de domadores de escarabajos húngaros, válidos para todos aquellos que vivan a más de 125 grados de latitud Oeste, según se mira desde Campuchea oriental).

— ¡No Obelix, no!, ya sabes que de pequeño, cuando todavía usabas dodotix, te caiste en una marmita llena de poción mágica, por lo cual los efectos son permanentes en ti. Tomar más podría ser peligroso.

Aquello enfadó bastante a Obelix, lo cual suele ser terrible (que se lo pregunten a los romanos). Olvidando su fuerza, le dio una tremenda patada al caldero de la poción, rompiéndolo en 8 trozos, algunos de los cuales se perdieron más allá del horizonte.

— Esto sí que es un problema —dijo Panoramix con evidente gesto de preocupación—. Sin la marmita mágica no es posible volver a preparar la poción.

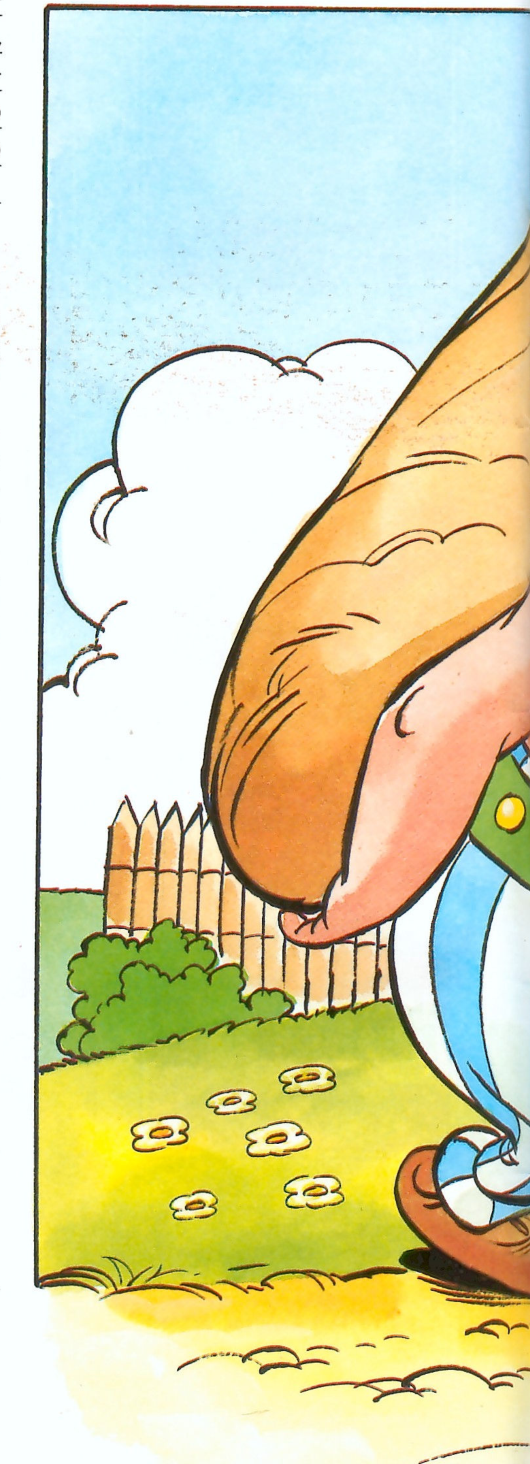
El pánico reinó entre la multitud, hasta que se escuchó la voz de Abraracurcix por encima de los aldeanos:

— ¡Que no nos invada el miedo!, recordad que la nuestra es una raza noble. Sólo debemos temer una cosa: que el cielo se caiga sobre nuestras cabezas. Si lográramos reunir todos los fragmentos, ¿no podría Esautomatix, el herrero, recomponer el caldero?

— Sí —contestó Panoramix con su locuacidad acostumbrada.

— Partid pues ahora mismo, Asterix y Obelix, y tratad de encontrar los trozos antes que los romanos se enteren de que no tenemos poción.

Un pequeño fragmento había quedado sobre las brasas, con un poco del mágico



brebaje en su interior. El druida lo guardó en una cantimplora y se lo entregó a Asterix con estas palabras:

— Es todo lo que queda, tan sólo una ración. Llévatela y úsala cuando lo creas conveniente. Adiós y que Tutatis os acompañe.

Una vez conocida la historia, pasemos al juego, que es de lo que se trata, ¿no?

Para jugar necesitáis un ordenador, el «Asterix and the magic cauldron» y un joystick (del latín joystico, joysticas, joysticare, joysticavi, joysticatum, según los traductores del cotolengo de San Molistrodemo).

Aunque el objetivo del juego es bastante difícil de lograr, el manejo, sin embargo, es extremadamente simple, lo cual en nuestra humilde, modesta y personal opinión es un serio punto a favor. El control se realiza en todo momento por medio del joystick, en las cuatro posiciones fundamentales (olvídate de las difíciles y odiosas diagonales), dando lugar a distintas acciones según en qué momento nos encontremos.

Si Asterix está en modo normal, es decir, andando por Roma, por el bosque, por los campamentos... o por donde sea, podéis dirigirle hacia arriba, abajo, derecha o izquierda, y hacerle beber la poción mágica pulsando dos veces seguidas el botón de disparo.

Si se encuentra luchando (en la «caja de combate») controláis el movimiento a derecha o izquierda, el salto y puedes hacer que se agache (Asterix es bajito, pero nunca viene mal para esquivar un ataque). En modo lucha y con el botón pulsado da puñetazos, hace barridos e incluso obsequia a los contrincantes con un fabuloso uppercut.

Una vez acabada la extensa y profunda



explicación sobre el manejo del joystick, pasemos a la descripción de la pantalla, que siempre es interesante.

La ventanita iluminada que tenéis frente al ordenador (si no la veis corred cuanto antes al oculista o reclamad en la tienda donde os vendieron el monitor, porque suena a estafa) está dividida en dos partes principales (la verdad es que sólo se divide en dos partes, pero diciendo «principales» la frase queda redonda. Retórica elemental).

— La zona inferior da una visión del escenario donde se desarrolla la acción, así como de los personajes que se encuentran cerca. Cuando Asterix entra en modo lucha, es aquí dónde aparece la «caja de combate» (ya explicaremos lo que es a su debido tiempo, no seáis impacientes).

— La superior es una banda de información que informa (esto no es retórica, es una chapuza) sobre la situación en cada momento, así como de los objetos que llevemos. Hay varios dibujitos, cada uno de los cuales supone un dato diferente:

1. El jamón. Como ya sabemos, Asterix y Obelix se alimentan casi exclusivamente de jabalíes (son jabalívoros, como dirían los expertos), y tienen que comer cada cierto tiempo. El número que hay a la derecha indica la cantidad de comida disponible. Cuando uno de los dos sienta hambre, el dibujito parpadeará, y las provisiones se verán reducidas. Si Obelix tiene hambre y no hay comida dejará de seguirnos, mientras que si se trata de Asterix, perderá una vida. Los jamones se reponen matando jabalíes y recogiéndolos una vez muertos. Sólo podremos llevar cinco a la vez (recordad que Asterix es un guerrero galo, no un trailler de transporte).

2. El caldero. Señala las piezas que ya han sido encontradas de las siete que debes recoger. Para averiguar cuántas faltan hay que recurrir a una difícil operación matemática denominada sustracción, la cual, dada su enorme complejidad temática y formal no detallaremos aquí (consúltense manuales de aritmética avanzada).

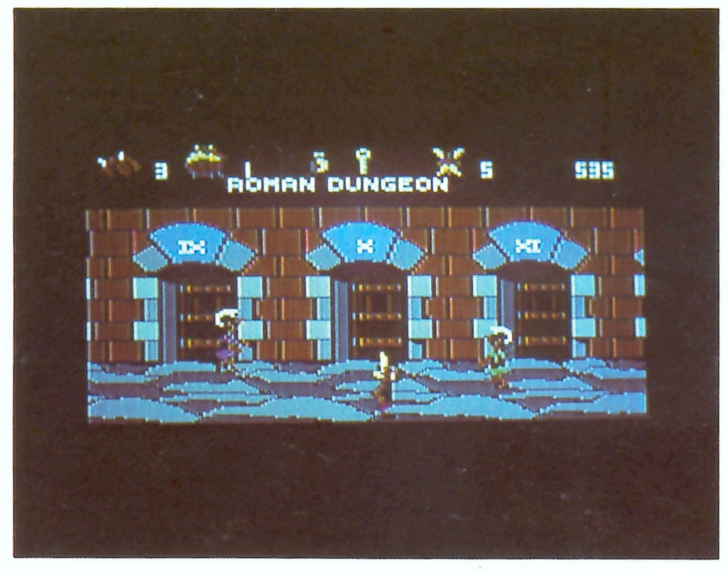
3. La poción. Este dibujo está en pantalla mientras no hayamos hecho uso del brebaje mágico. Recordad que sólo disponéis de una dosis, así que estudiad bien el momento de utilizarla y no la malgastéis. Cuando la bebáis, el indicador parpadeará. En el momento en que desaparece, haceos a la idea de que no tenéis poción ni demasiadas posibilidades de acabar con éxito, pero qué le vamos a hacer, así es la vida...

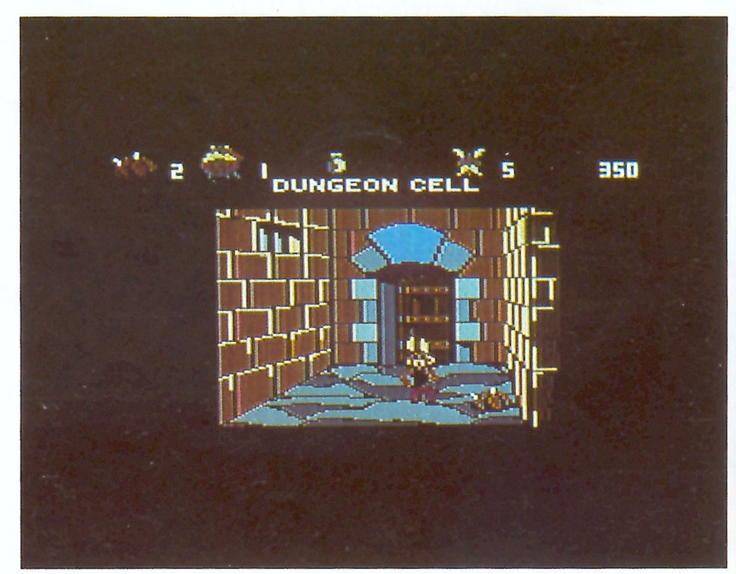
4. La llave. Indica si Asterix lleva alguna idem encima. Las «idem» no le sirven para nada, pero las llaves le permiten pasar por las puertas cerradas con idem. Si llegáis a una idem cerrada con llave, y no disponéis de una idem para abrirla, no podréis pasar, por lo cual es muy útil llevar siempre encima alguna llave para las puertas cerradas con idem (claro, ¿no?).

5. Asterix. Señala el número de vidas que quedan. Cuando llega a cero, se acabó la partida. Empezamos con cinco, y por mucho que le lloremos y le supliquemos al ordenador no nos dará ni una más. ¡Si será egoista!









6. Puntuación. Pues ni más ni menos que eso. ¿Qué esperabais?

7. Situación. Es un textito que da idea del lugar donde nos encontramos. No es una tontería: resulta más práctico de lo que parece para moverse de un lado a otro. El fatídico y descorazonador «Game Over», que conocemos y odiamos con toda la fuerza de nuestras caderas, aparece aquí, cuando el game está over.

Caminando, luchando, recogiendo cosas y (confesémoslo) perdiéndonos por la Galia

En todos los juegos (bien lo sabemos) hay cosas fáciles y difíciles de realizar, pues bien, hay algo que es increiblemente sencillo hacer en éste: perderse. Recordemos que en aquellos tiempos la Galia distaba mucho de ser la primera potencia en cuanto a producción de brújulas se refiere.

Para conseguir todos los trozos del caldero, debéis recorrer el bosque, la aldea, los campamentos fortificados romanos y llegar incluso a Roma, que se encuentra cerca del poblado (craso error geográfico, pero lo pasaremos por esta vez). La dificultad para orientarse no estriba en el tamaño de la zona, sino en la disposición de las pantallas. Estas no se relacionan entre sí de un modo biunívoco, aunque la secuencia y el orden de las mismas es siempre igual.

— Oye, eso no ha quedado nada claro. Explícalo de nuevo.

— Está bien, está bien, ahora iba a hacerlo.

— Pues venga, que se vea.

— (¡Puff!, vaya jefe de redacción).

Como íbamos diciendo, si Asterix se encuentra en una pantalla A y se mueve hacia la derecha, llega a otra pantalla B. Si vuelve por el mismo camino, es decir, estando en B se desplaza hacia la izquierda, no siempre regresa a A, sino que, según en que zonas, aparece en una nueva pantalla C. Esto que puede parecer un obstáculo insalvable a la hora de confeccionar un mapa, no lo es tanto si tenemos en cuenta que el orden de las pantallas es siempre el mismo.

En el ejemplo anterior, al salir de B siempre llegamos a C, pero nunca volvemos a A ni aparecemos en otra diferente D. Por este motivo, es posible ir realizando mapas de las zonas por las que pasemos, aunque resulte un poco más laborioso que en otro tipo de juegos (¿a qué esperáis?. Coged lapiz y papel y empezad cuanto antes).

— ¿Qué te parece así?

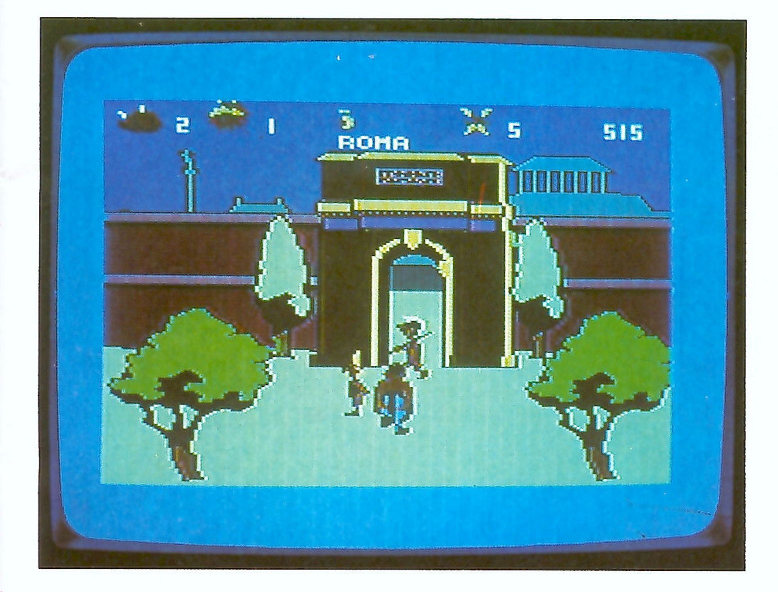
— Sí, sí, ahora se entiende mucho mejor. Puedes seguir.

— Muchas gracias.

— No hay de qué.

Además, los fragmentos del caldero están siempre en el mismo sitio (muy agradecido, señor programador), así que, una vez localizados, no debéis tener demasiados problemas para reunirlos todos.

Tal vez la zona más difícil de todo el juego sea Roma, dada la cantidad de legionarios y centuriones que tratarán de acabar con nosotros. «Están locos estos romanos»









(Obelix, «Roma y yo», volumen 7, pág. 483).
De todos modos, no hay más remedio que entrar en la ciudad, si queréis conseguir una llave o varias piezas del caldero que se encuentran aquí.

Durante toda la partida, controláis a Asterix, mientras que su rellenito amigo se limita a seguirle. Obelix, apesadumbrado y deprimido por ser el causante de todo este embrollo, no tiene ganas de hacer nada, así que hace exactamente eso: nada. No contéis con su ayuda si os atacan los romanos o para salir de alguna situación apurada. Incluso al llegar a Roma, no nos seguirá, quedándose en la puerta a esperarnos, quizá porque «los romanos son como las ostras: un poco está bien, pero si abusas te pueden llegar a sentar mal» (Obelix, «El arte del buen comer y del buen zurrar», volumen 4, pág. 216).

Cuando Asterix se cruza con algún enemigo (jabalíes, gladiadores, legionarios o centuriones), aparece la caja de combate, en la que están situados los dos contrincantes, y al lado de cada uno una banda vertical que refleja la potencia de que dispone. La fuerza varía dependiendo del enemigo (desde el jabalí, más débil, pasando por legionarios y centuriones hasta el gladiador, que es una mala bestia con coraza), de los golpes recibidos, de si se ha bebido o no la poción, y de la comida ingerida.

Estando en la caja de combate, los luchadores deben golpearse entre sí, hasta que la franja de energía de uno de los dos llegue a cero, lo que indicará que ha muerto. En modo lucha, no podéis beber poción, así que no entréis a pelear sin pensároslo primero.

De todos modos, si veis las cosas muy feas, siempre os queda la posibilidad de escapar. Esta no es una medida demasiado valiente, pero como dijo hace tiempo Napoleón: «Una retirada a tiempo es una victoria». (Recordemos que Asterix y Napoleón tienen un par de cosas en común: la nacionalidad y la estatura). Para salir de la caja de combate es necesario haber golpeado como mínimo una vez a nuestro enemigo (pacifistas abstenerse), alejarnos hacia el extremo de la caja, manteniendo el joystick en esta posición durante dos segundos, y a correr.

Disponéis además de una tercera posibilidad que no está contemplada en las instrucciones: rendiros. Si nada mas aparecer la caja de combate permanecéis quietos, sin hacer el mínimo movimiento, en algunos casos el legionario o el centurión (sólo sirve con estos dos) os detendrán, e iréis a parar a las mazmorras de Roma. A veces esto no funciona y el enemigo se dedica a machacarnos a palos, pero hay que probar. (Un consejo de amigo: no lo intentéis estando en Roma).

Una vez en los calabozos podéis hacer dos cosas:

1. Si no tenéis la llave, o teniendo la idem no os acercáis a la puerta (que es el modo de usarla), al pasar un cierto rato seréis conducidos a la arena del circo, donde deberéis luchar con un gladiador. Procurad vencerle, porque una vez muerto, descubriréis que posee una de las piezas que buscáis. Cuando haya acabado el combate podréis salir a las calles de Roma.

2. Si tenéis la idem, pegaos a la puerta, con lo que saldréis del calabozo y podréis entrar en todos los demás, pues la llave vale para todas las puertas. En las mazmorras podréis encontrar galos presos, romanos cumpliendo condena y simplemente cárceles vacías, pero atención a tres puertas en especial:

— En la I hay un gladiador que no os atacará si no le provocáis, pero de todos modos no es necesario pasar a hacerle una visita, ya que éste no tiene ningún fragmento del caldero.

— La puerta VI es una salida a Roma.

— Y en la III (¡¡Tachán, tarara chan!!) encontraréis un trozo del caldero que vais buscando. Sería interesante un estudio sobre cómo ha llegado a parar allí (¿qué hace una pieza como tú en un calabozo como éste?), pero sinceramente, hoy no tenemos ganas de pensarlo. Trabajad en el tema, y el Nobel de inutilogía os lo lleváis este año; fijo.

A parte de todo lo explicado anteriormente, la caja de combate aparece también cuando estáis cerca de un objeto que debéis recoger (jamones, manzanas, llaves, oro y pedazos de caldero), para lo cual basta con situarse sobre él. Fácil, ¿no?

VEREDICTO FINAL

GRAFICOS	****
SONIDO	****
ORIGINALIDAD	**
DIFICULTAD	****
INTERES	****

Asterix y el Caldero Mágico: no es la quintaesencia de la informática, pero no está nada mal, no señor.

La simplicidad de manejo hace que resulte un juego entretenido, y no agobiante, como tantos que vemos aparecer últimamente. La facilidad de movimientos y el hecho de encontrar las piezas en el mismo sitio siempre, consiguen que no sea demasiado difícil, así como el trazado del mapa y lo duro de los combates logra que no sea demasiado fácil. En nuestra opinión, el equilibrio está admirablemente bien conseguido, con un desarrollo de lo más entretenido.

En cuanto a gráficos, no hay más que ver las fotos. Sin ser una maravilla en alta resolución (no os esperéis una reproducción de «La Gioconda» de fondo, porque os desilusionaréis), son muy nítidos y vistosos, jugando fantásticamente bien con las combinaciones de colores (si tenéis un monitor en blanco y negro es una verdadera lástima).

La música es, tal vez, de lo más pegadizo que hayáis oído a vuestro Commodore en mucho tiempo. Si sois capaces de jugar unas cuantas partidas seguidas y no sorprenderos a vosotros mismos tarareando la melodía, escribidnos y decidnos cómo lo habéis hecho.

En resumen: un juego entretenido, con gráficos y sonido más que aceptables; no está de más tenerlo. ¡Está muy bien, por Belenos!

FICHA TECNICA

NOMBRE	ASTERIX AND THE MAGIC CAULDRON
PRECIO	2.300 PTAS.
SOPORTE	CINTA
TIPO	ACCION
MODELO	C-64, C-128
OBSERVACIONES	NINGUNA

LOS EXITOS DEL CINE
EN TU ORDENADOR

C

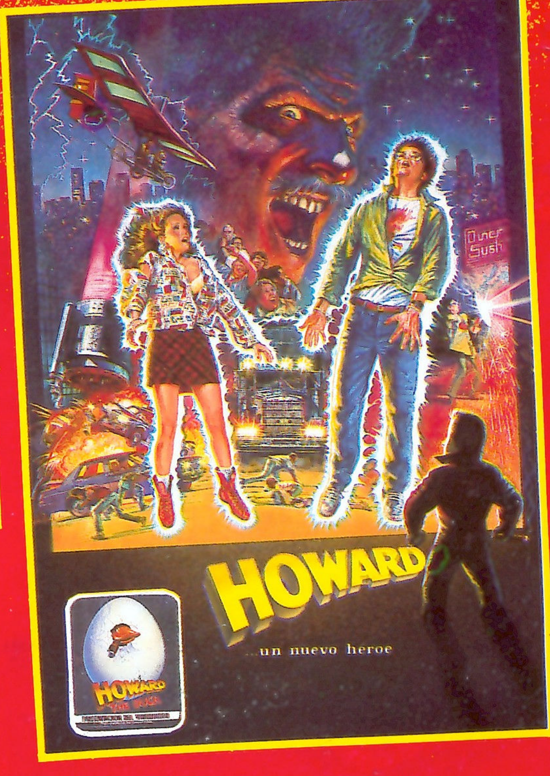


CSA.





CSA.



CSA.

Disponibles para:

COMMODORE C
SPECTRUM S
AMSTRAD A
CASS / DISK

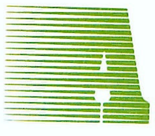
EN TIENDAS ESPECIALIZADAS Y GRANDES ALMACENES, O DIRECTAMENTE POR CORREO O TELEFONO A: **PROEIN, S.A.**

Distribuido en Cataluña por: DISCOVERY INFORMATIC C/. Arco Iris, 75 - BARCELONA - Tels. 256 49 08 / 09

Velázquez, 10 - 28001 Madrid - Tels. (91) 276 22 08/09

Código Máquina para la 1541

En estos artículos vamos a intentar manejar la unidad de disco y los datos almacenados en código máquina. Para ello, es indispensable conocer el BASIC y el lenguaje ensamblador del microprocesador 6510; con idea de hacerlo más pedagógico intentaremos siempre que sea posible transcribir su equivalencia en BASIC.



unque el BASIC puede solucionar la gran mayoría de nuestras necesidades en el tratamiento de ficheros, debemos considerar el caso de que manejemos volúmenes muy grandes de información; aquí es donde se nos revela el código máquina como indispensable.

En principio no nos debe asustar el lenguaje máquina y debemos pensar que hay casos en los cuales es la única solución viable, si no queremos sucumbir ante nuestra pantalla.

Lógicamente, cualquier instrucción que le podamos mandar a la unidad de disco en BASIC, la podemos simular en código máquina, y esto es lo que veremos a continuación.

Vamos a aprender cómo abrir un fichero para lectura o escritura, cerrar el fichero, validar un disco, formatear el disco o interrogar el canal de errores de la unidad, tras cualquier orden enviada a la unidad.

Para todas estas operaciones nos valdremos de las rutinas KERNAL contenidas en la ROM del 64. Dichas rutinas están implementadas en el C-64 y otros ordenadores Commodore para facilitar la conversión de programas en código máquina de un ordenador a otro. A estas rutinas se accede a partir de una tabla de saltos standard ubicada en la última página de memoria del ordenador.



Como ayuda a la programación en C/M, hemos utilizado el «Machine Lighting» (macroensamblador de dos pasadas de la casa Oasis Software), aunque para este cometido puede servirnos cualquier otro monitor o ensamblador de código máquina. Para aquellos que no dispongan de ninguno, introducimos cargadores en BASIC, los cuales han sido realizados con el generador de datas automático, aparecido en nuestra revista número 4, primera época.

Cómo abrir un fichero en código máquina

Muchas operaciones con la unidad de disco se pueden obtener de diversas formas, pero para la gran mayoría de ellas, es necesaria la apertura de un fichero.

Este cometido lo logramos a través del KERNAL, las rutinas a emplear son la SETNAM, SETELFS y OPEN propiamente dicha.

SETNAM ($FFBD)

Nos sirve para definir el nombre del fichero, pero debe ser llamada aunque no necesitemos el nombre del fichero, en cuyo caso ajustaremos la longitud del nombre a cero.

Para poder utilizarla, debemos comunicarnos con el ordenador a través de los registros del microprocesador de la siguiente manera:

1. Cargamos el registro A (acumulador) con la longitud del nombre del fichero.
2. Cargamos el registro X (índice x) con el byte bajo de la posición de memoria donde empiezan los caracteres ASCII el nombre del fichero.
3. Cargamos el registro Y (índice y) con el byte alto de la posición de memoria donde empiezan los caracteres ASCII del nombre del fichero.
4. Saltamos a la subrutina con JSR $FFBD, lo que equivale a SYS 65469.

SETELFS ($FFBA)

Esta rutina del KERNAL sirve para ajustar el número de fichero lógico, el periférico afectado y la dirección secundaria de la instrucción OPEN.

Para empleo nos comunicaremos con los registros de la siguiente manera:

1. Se carga el registro A (acumulador) con el número de fichero lógico (número del 0 al 255).
2. Se carga el registro índice X con el número de dispositivo periférico.
3. Se carga el registro índice Y con el número de la dirección secundaria deseada (canal de comunicación o buffer de la unidad); si abrimos el canal para enviar una instrucción o verificar el «status» de la unidad, debemos utilizar la dirección secundaria 15 ($ØF).
4. Se salta a la subrutina con JSR$FFBA o lo que es lo mismo SYS 65466.

OPEN ($FFCØ)

Esta es la rutina del KERNAL que propiamente abre el fichero, para lo cual debemos previamente haber accedido a SETNAM y SETLFS. Aunque el orden de utilización de estas dos últimas no es significativo, sí es importante que su ejecución sea anterior a la de OPEN.

Cuando llamamos a esta rutina, los valores contenidos en los registros A, X e Y son



destruidos, por lo que si tenemos algo en ellos que nos hace falta, debemos guardarlo previamente en el stack o en alguna posición de memoria.

En este caso, no debemos comunicarnos con el microprocesador a través de ningún registro, tan sólo tenemos que hacer el salto a la subrutina con un JSR$FFCØ o con un SYS 65472.

Si después de utilizar esta rutina, con el acarreo desactivado, vuelve activado, significa que ha habido un error, cuyo código figurará en el acumulador, según el siguiente significado:

- #0 parada mediante STOP
- #1 demasiados ficheros abiertos
- #2 fichero abierto
- #3 fichero no abierto
- #4 fichero no encontrado
- #5 periférico no presente
- #6 fichero no de entrada de datos
- #7 fichero no de salida de datos
- #8 nombre de fichero perdido
- #9 número de periférico ilegal
- #242 RS232 OPEN/CLOSE

Utilización de conjunto

Con estas sencillas rutinas que hemos visto, tenemos acceso a una gran cantidad de comandos del disco, los cuales corresponden al formato BASIC OPENa,b,c"texto,d,c,":CLOSEa.

Como ejemplos, podemos comprobar el efecto de:

OPEN1,8,15,"VØ": valida un disco; la dirección secundaria es 15 ya que es un comando del disco.

OPEN1,8,15,"DATA,S,W,,: prepara un fichero llamado DATA de tipo secuencial para la escritura.

OPEN1,8,15," Ø:MOD VØ2,P,W": escribe con remplazamiento un fichero de programa llamado MOD V02.

OPEN1,8,15,"NØ:TEST/DEMO,01": formatea un disco con el nombre TEST/DEMO y el identificador lógico de disco 01.

OPEN1,8,15,"|# 2": abre un fichero de acceso directo al disco a través del buffer # 2.

Todas estas instrucciones se pueden simular en código máquina con SETNAM SETLFS y OPEN.

Rutina ejemplo en código máquina

Como aplicación de lo visto hasta ahora, formatearemos un disco con el nombre de TEST/DEMO y el identificador lógico 01.

Para ello, primero tenemos que ubicar en la memoria del ordenador la cadena de caracteres ASCII que corresponde al nombre y al identificador (incluida la coma de separación). A tal fin, podemos usar POKEs o una rutina en código máquina; en nuestro caso, vamos a suponer que ya la tenemos en memoria y está situada en el buffer del cassette ($0C3C), y la longitud del texto es de 12 caracteres ($0C).

```
LDA # $OC  ;  longitud del texto ASCII.
LDX # $3C  ;  byte bajo de la situación
              del texto ($0C3C).
LDY # $0C  ;  byte alto de la situación
              del texto ($0C3C).
JSR $FFBD  ;  llamada a SETNAM.
LDA # $01  ;  fichero lógico número 1.
LDX # $08  ;  número de periférico 8.
LDY # $0F  ;  dirección secundaria 15.
JSR $FFBA  ;  llamada a SETLFS.
JSR SFFC0  ;  llamada a OPEN.
Texto ASCII en $OC3C TEST/DEMO,01
```

No son necesarias las comillas del BASIC. Todavía no sabemos cerrar un fichero, pero lo veremos más adelante, como ocurre en el BASIC no debemos dejar ficheros abiertos tras ser utilizados.

Lectura de un fichero en código máquina

Como en el caso de la apertura de ficheros, existen una serie de rutinas del KERNAL que están especialmente pensadas para la lectura de ficheros, los cuales deben estar convenientemente abiertos y designados como ficheros de lectura.

Las rutinas a utilizar son CHKIN (prepara un canal para input de datos al ordenador), CHRIN (lee un carácter del canal de entrada), CLRCHN (limpia el canal de entrada/salida para evitar errores) y GETIN (tomar un carácter del buffer del teclado o de la interface RS232).

Rutina CHKIN ($FFC6)

Sirve para designar un canal o fichero lógico previamente abierto con OPEN, como canal de entrada de datos al ordenador; o dicho de otra manera, se utiliza para desviar la entrada del teclado a la de un fichero (por ejemplo, leer datos del disco y colocarlos en unas posiciones de memoria determinadas).

Esta rutina tiene algunas particularidades. Como es lógico, el periférico al que nos referimos debe ser capaz de entregar datos al ordenador, ya que en otro caso, nos daría error y la rutina sería abortada. Cuando se produce un error, se activa la bandera de acarreo del registro de estado, por lo que es conveniente desactivarla antes de llamar a esta rutina para poder testearla sin problemas posteriormente. Los códigos de error aparecen en el acumulador y son los mismos que se usaban en la rutina OPEN.

Si queremos utilizar el teclado como dispositivo de entrada y no hay otros ficheros abiertos previamente, no es necesario llamar a esta rutina antes de poder usar CHRIN o GETIN, tampoco se hace necesario en este caso llamar antes a OPEN.

Debemos tener cuidado con los datos que tenemos en el registro X y el A, ya que tras llamarla quedarán destruidos.

Para usar esta rutina tenemos que hacer lo siguiente:

1. Cargar el registro índice X con el número del fichero (previamente abierto y definido), que queremos usar como canal de entrada de datos.
2. Llamar a esta subrutina con un JSR $FFC6 o con SYS 65478.

Como es lógico, es opcional el comprobar los errores después, o desactivar la bandera de acarreo antes de llamar a esta subrutina.

Rutina CHRIN también llamada BASIN ($FFCF)

Se utiliza para leer caracteres desde un canal de entrada de datos abierto y designado como canal de entrada mediante CHKIN. Esta rutina cada vez que es llamada, va tomando un carácter del canal y lo va colocando en el acumulador, desde éste podemos situarlo donde queramos dentro de la memoria disponible del ordenador, mediante el direccionamiento indexado. El canal queda abierto tras la llamada a la subrutina.

Los datos contenidos en los registros X y A son destruidos tras llamar a esta rutina, por lo que no podemos usar el índice X para almacenarlos en la memoria, teniendo que emplear forzosamente el índice Y para ello.

Uso de CHRIN

1. Abrir fichero.
2. Designarlo como fichero de INPUT (se omite en caso de teclado).
3. Colocar puntero en su valor Y.
4. Saltar a CHRIN.
5. Almacenar el contenido del acumulador indexando con Y.
6. Incrementar el puntero.
7. Ver si es el último carácter a obtener.

Ejemplo:
Leer desde el fichero record # 15 hasta encontrar un retroceso de carro (CR).

```
     LDA # $03    ; longitud del nombre.
     LDX # $LOW ; posición del nombre.
     LDY # $HI    ; posición delnombre.
     JSR $FFBD   ; salto a SETNAM.
     LDA # $01    ; fichero.
     LDX # $08    ; periférico.
     LDY # $0F    ; dirección secundaria.
     JSR $FFBA   ; salto a SETLFS.
     JSR $FFC0   ; salto a OPEN.
     LDX # $01
     JSR $FFC6   ; salto a CHKIN.
     LDY # $00    ; inicializa puntero.
READ JSR $FFCF   ; obtiene un carácter.
     CMP # $0D   ; comprueba si es
                   return.
     BEQ END     ; salto a final.
     STA $0800,Y ; almacena carácter.
     INY         ; incrementa puntero.
     JMP READ    ; lee otro carácter.
END  JSR $FFC0   ; restaura I/O.
```

NOTA: Al acabar debemos cerrar el fichero, pero eso lo veremos más tarde.

Rutina CLRCHN ($FFCC) 65484

Cuando acabamos una I/O debemos llamar a la rutina CLRCHN. Esta desconecta el fichero seleccionado por CHKIN como fichero de input y selecciona las I/O normales (pantalla y teclado). Como dato importante, debemos señalar que CLRCHN, no cierra ningún fichero, tan sólo lo deselecciona como I/O temporalmente; es importante no olvidarla, ya que nos daría problemas. Para usarla, tan sólo debemos saltar a la posición $FFCF con un JSR o a 65484 con SYS, destruyendo lo que ha construido en el registro X y A.

Escritura de un fichero en CM

Como en el caso de la lectura, hay una serie de rutinas en C/M correspondientes al KERNAL para la escritura de ficheros. Estas son CHKOUT, (como CHKIN, selecciona un fichero para canal de salida), CHROUT (escribe un carácter) y CLRHN (como en el caso de la lectura limpia el canal de I/O).

El modo de uso es similar a la lectura de datos desde los periféricos de input.

Rutina KERNAL CHKOUT $FFC9 (65481)

Esta es la primera rutina KERNAL que debemos usar para indicar que un determinado fichero es de salida; como en el caso de CHKIN éste debe estar previamente abierto.

El equivalente BASIC es CMD y los errores que pueden aparecer se devuelven por el acumulador, activándose con anterioridad el acarreo. Los códigos son los mismos

que en CHKIN; así por ejemplo, si intentamos escribir en un fichero de lectura, aparecerá el código # 7 «not output file» previa activación del Carry Flag en el registro de estado. Para su utilización, es preciso cargar el registro X con el fichero deseado y hacer JSR $ FFC9 o SYS 665481. Destruye lo que hay en X y A, por lo que se deben tomar las precauciones pertinentes, si el contenido de dichos registros nos es de utilidad.

Rutina KERNAL CHRONT $FFD2 (65490) BSOUT

Esta rutina es la que realmente escribe un carácter en un fichero de salida, si no se ha definido un fichero de salida con CHKOUT, el dato es enviado a la pantalla; afecta los registros X y A, por lo cual no es posible usar el X como índice.

Proyecta el ASCII a la pantalla si hacemos:

```
LDA# $41    POKE 1024,1
JSR BOSOUT
```

siendo posible enviar tanto caracteres como código de control o color.

```
PRINT # X CHR$(X)
```

Si utilizamos previamente CHKOUT, el carácter no va a la pantalla, sino al fichero especificado.

CLRCHN $FFCC

Limpia el canal de I/0, al igual que cuando se abre un fichero para lectura. Así por ejemplo, escribir los 16 bytes desde $0900 en el fichero # 1, supone:

```
        LDX # $01
        JSR $FFC9
        LDY # $CF
LOOP    LDA $0800,Y
        JSR $FFD2
        DEY
        BNE LOOP
        JSR $FFCC
```

Cerrar un fichero

Existen dos rutinas que aparentemente cierran los ficheros: CLOSE $FFC3 y CLALL $FFET; nosotros sólo podemos utilizar close, ya que CLALL cierra todos los ficheros, pero internamente en el C-64 y no en la 1541, cosa que no ocurre con la instrucción CLOSE.

Instrucción CLOSE $FFC3

Esta rutina KERNAL cierra un fichero lógico después de una operación I/O. Para utilizarla debemos cargar el acumulador con el número de fichero lógico que queremos cerrar (igual que hacíamos con OPEN).

Esta rutina afecta a los registros A, X e Y y no devuelve ningún código de error. Si cerramos un canal RS-232 ocurren cosas especiales, el buffer queda liberado y se efectúa una instrucción CLR. Por otra parte, hay que tener cuidado, porque si cerramos un fichero no abierto, no se comunica ningún error.

Veamos algún ejemplo: cierre y limpieza del canal # 5.

```
JSR $FFCC; Restaura I/O.
LDA # $052
JSR $FFC3; Cierra el canal # 5.
```

Consulta del canal de errores en CM

La consulta del canal de errores sirve para:

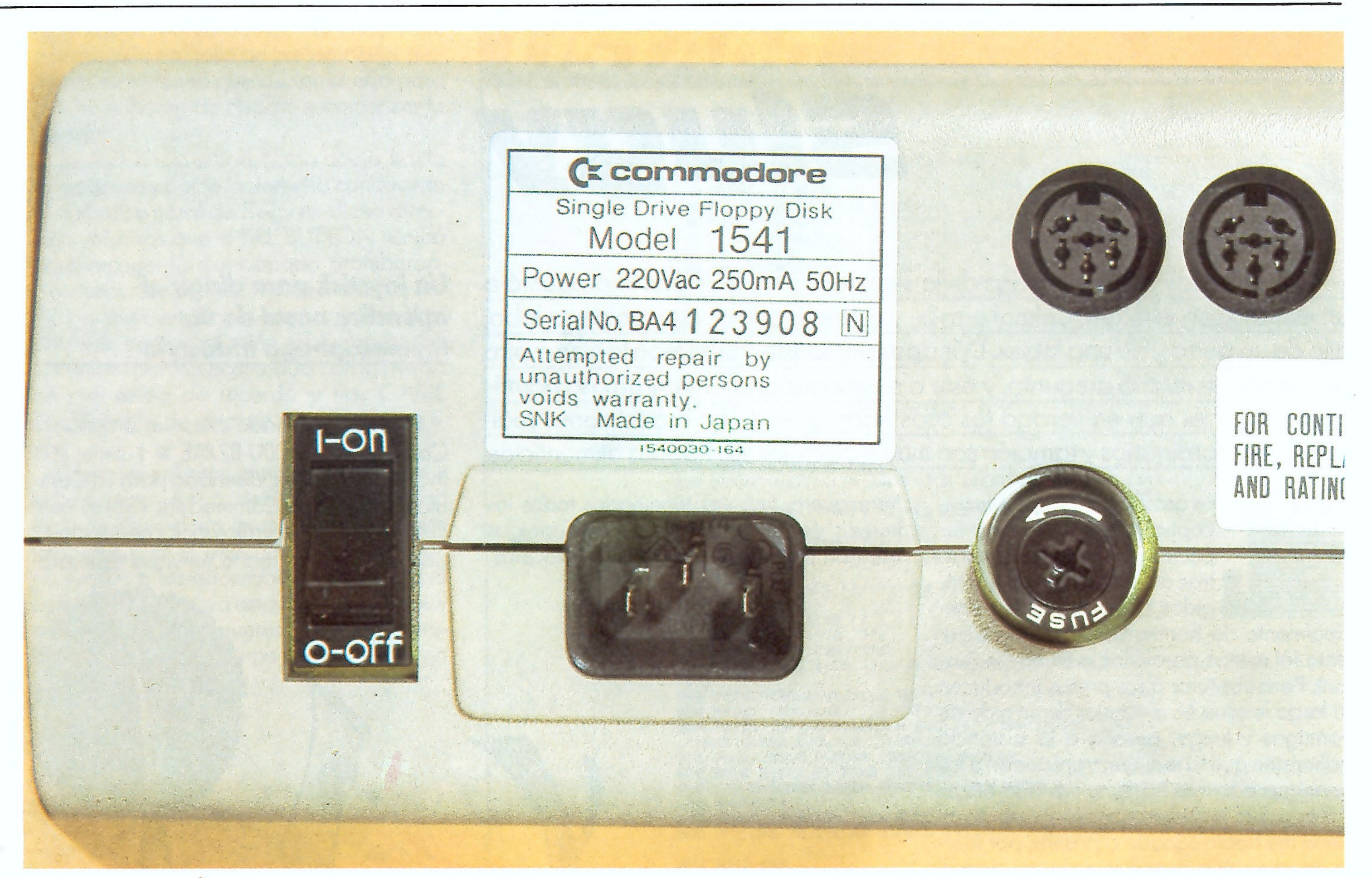
1. Ver si se han producido errores en el manejo del disco.
2. Encontrar la marca de fin de fichero.

Para lograrlo en BASIC, empleamos la variable reservada ST y en código máquina la rutina KERNAL READST $FFB7.

Rutina KERNAL READST ($FFB7)

Nos devuelve el «status» corriente en el acumulador de la siguiente forma:

$00 Todo bien.



$01 Time out on write.
$02 Time out on read.
$40 Fin de fichero.
$80 Dispositivo no presente.

Así por ejemplo, el programa para la lectura de fichero y salto si el dispositivo no está presente, sería:

```
        LDA #$03      ; longitud del
                        nombre.
        LDX #$LOW     ; posición low.
        LDY #$HI      ; posición hi.
        JSR $FFBD     ; salto a SETNAM.
        LDA #$01      ; fichero.
        LDX #$08      ; periférico.
        LDY #$0F      ; dirección secundaria.
        JSR $FFBA     ; salto a SETLFS.
        JSR $FFCØ     ; salto a OPEN.
        JSR SFFB7     ; salto a READST.
        COMP #$80     ; comprueba si está
                        presente.
        BEQ END       ; si no lo está acaba.
        LDX #$01
        JSR $FFC6     ; salta a CHKIN.
        LDY #$00
READ    JSR $FFCF     ; salta a CHRIN.
        CMP #$0D      ; comprueba si es
                        return.
        BEQ END       ; si lo es acaba.
        STA $0800,Y   ; lo almacena indexa-
                        do por Y.
        INY
        JMP READ
END     JSR $FFCC       salta a CLRCHN.
        LDA #01
        JSR $FFC3       salta a CLOSE.
```

Y esto ha sido todo, esperamos que sea útil y que sirva para aclararos las ideas con respecto al KERNAL y al código máquina del 6510.

Slugy

Cuentan las viejas leyendas de la selva que un día el león le preguntó a un extraño ser: «¿tú qué animal eres?», y éste respondió: «soy un perro lobo, hijo de un perro y de una loba». Dándose por satisfecho, el león le hizo a otro ser extraño la misma pregunta, y éste a su vez respondió: «soy un oso hormiguero,...». Y es que en verdad los osos hormigueros son especímenes realmente extraordinarios y también son muy especiales sus hábitos alimenticios.

Los osos hormigueros (Mymecophaga tridactyla) son especies de mamíferos xenartros de la familia mirmecofágidos, que se alimentan principalmente de hormigas, un manjar suculento (al menos para ellos, a mí no me gustan). Para capturar a sus presas introducen su larga lengua en el interior de un nido de hormigas y luego, gracias a la sustancia adherente que la recubre, transportan a las hormigas o termes hasta su cavidad bucal.

Todo esto lo cuento porque el otro día mi perro me había sacado a pasear, por suerte, sin correa (la correa siempre me deja el cuello dolorido). Pues bien, correteaba yo de aquí para allá distraidamente cuando de pronto un oso hormiguero me pidió una linterna. «¿Para qué?», dije. Ante lo cual él se encogió de hombros y desapareció.

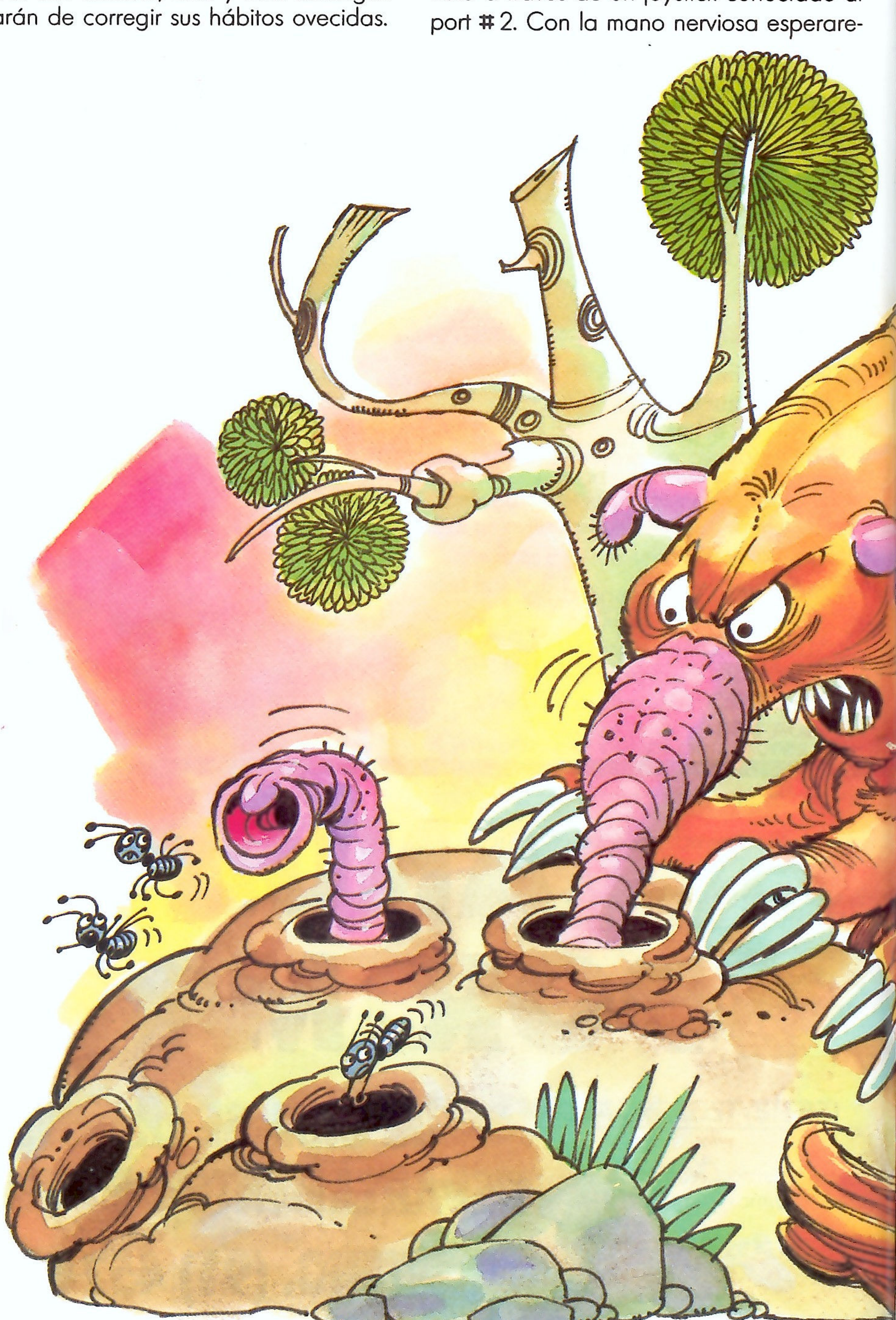
Acto seguido mi perro me subió a casa, dando por terminado el paseo, pero a mí ya no me importó en absoluto, mi mente trabajaba a un ritmo desenfrenado: a) padezco «delirium tremens», b) estoy loco, o c) realmente un oso hormiguero me había pedido una linterna. Por si acaso se trataba de una señal para que yo reivindicara los derechos del oso hormiguero y como en aquel instante me urgía encontrar una idea para un programa, decidí crear este juego, cuyo título es SLUGY.

Slugy es un simpático oso hormiguero cuya única preocupación es la de conseguir comida. Desgraciadamente, por raro que parezca, Slugy es alérgico a las hormigas y tan sólo puede alimentarse de los huevos de estas criaturas. Así, si una hormiga toca la nariz de Slugy (no tiene lengua y por ello debe introducir toda su nariz en el hormiguero), este lo verá todo gris, su lengua se volverá rosa y, al cabo de tres contactos, morirá.

Comienza Slugy su manutención diaria en un hormiguero donde las hormigas están totalmente desprevenidas y apenas sí puede tropezarse por despiste con alguna de ellas. Pero a medida que pase a otro hormiguero, una vez terminados todos los huevos del anterior, más y más hormigas tratarán de corregir sus hábitos ovecidas.

Un joystick para dirigir el apéndice nasal de un mymecophaga tridactyla

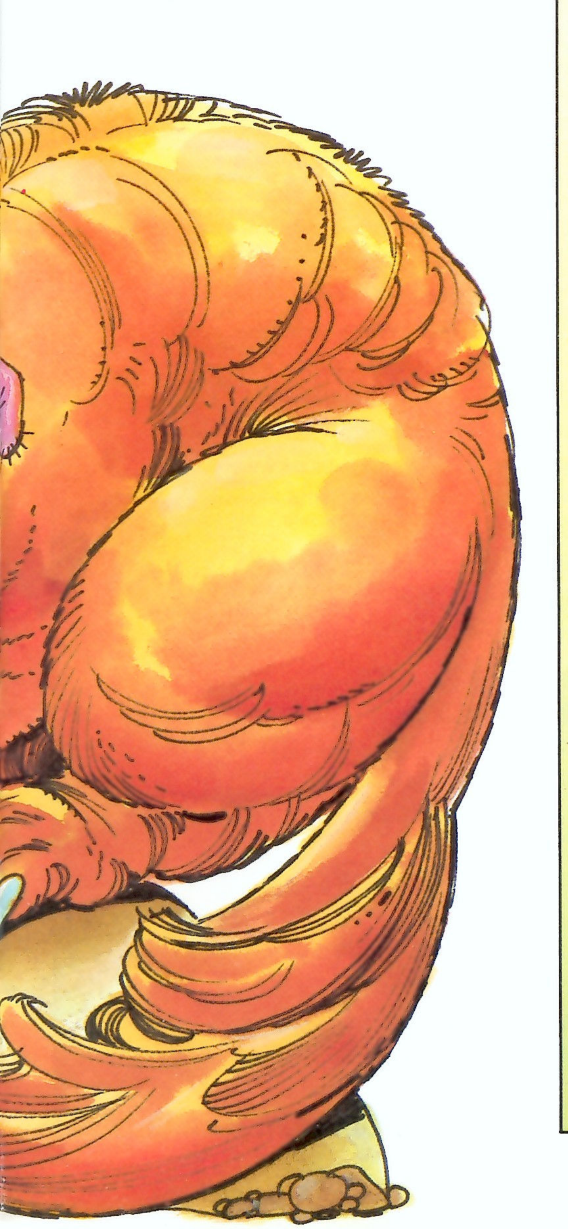
El juego, diseñado para un ordenador Commodore 45000-87/AE # 1 serie 708 (no, es una broma, diseñado para un Commodore 64 o un Commodore 128) se controla a través de un joystick conectado al port # 2. Con la mano nerviosa esperare-



mos después del consabido RUN hasta que aparezca la pantalla de presentación, momento más idóneo que cualquier otro para oprimir el botón de disparo y comenzar la partida.

Joystick hacia arriba, hacia abajo, hacia la izquierda y hacia la derecha conducirán el apéndice nasal de Slugy en dicha dirección, mientras que el FIRE BUTTON servirá para recoger la mencionada protuberancia nasal. Salvo estas indicaciones lo demás es bien sencillo: al acabar, tras haber obtenido una puntuación suficiente, graba tu nombre para la posteridad con la ayuda de una rutina de records; y tras GAME OVER, botón de disparo y vuelta a empezar.

Como veis el programa es muy simple y el listado muy largo, debido a la cantidad de instrucciones en lenguaje máquina de las cuales consta. ¡Ah!, otra cosa. Por favor, no llaméis a los esforzados hombres de la camisa de fuerza, ya estoy bien, no he vuelto a ver osos hormigueros... aunque la verdad es que hace tiempo que mi perro no me saca a pasear.



LISTADO

```
10 PRINTCHR$(142);CHR$(8):GOTO 610                     -230-
100 POKE 53281,0                                       -095-
110 PRINT"{CLR}{BLK}";:                                -027-
120 PRINT"{RON}{ACL}[40 ESP]";                         -107-
130 PRINT"{RON}{ACL}[40 ESP]";                         -108-
140 PRINT"{RON}{ACL}[40 ESP]";                         -109-
150 PRINT"{RON}{ACL}[40 ESP]";                         -110-
160 PRINT"{RON}{ACL}[40 ESP]";                         -111-
170 PRINT"{ROF}{NRJ}*++[3 *]+**[3 +]**++*+.{ACL}{RON}
{CYN} {ROF}{NRJ},*+**+*+*+[3 *]+[5 *]+";               -208-
180 PRINT"{CYN}[40 ']";                                -124-
190 PRINT"{NRJ}[4 (][3 )](()()(([5 )][9 (]))[5 (][5 )
]";                                                    -152-
200 PRINT"{NRJ}*+*[4 +]**[6 +][3 *]++*+**+*+*[4 +]*+[
3 *]+**";                                              -229-
210 PRINT"{CYN}[40 ']";                                -118-
220 PRINT"{NRJ}[5 (][3 )]()[3 (][3 )][4 (])(()()(()((
)[4 (][4 )]";                                          -145-
230 PRINT"{NRJ}*++[6 *][4 +]*+*+*[6 +][3 *]+[4 *]+*+[
4 *]+";                                                -228-
240 PRINT"{CYN}[40 ']";                                -121-
250 PRINT"{NRJ})(([3 )][7 (][6 )]())(()(()()()[3 (]))
())";                                                  -152-
260 PRINT"{NRJ}*+*+[4 *]+*[6 +][4 *]+*+[6 *]+*+*+*+
**";                                                   -229-
270 PRINT"{CYN}[40 ']";                                -124-
280 PRINT"{NRJ}()[4 (]))()(([4 )](()())()[4 (])(()()(
[3 )]((";                                              -153-
290 PRINT"{NRJ}[3 *]++*+*+[4 *][4 +][3 *]+*+*+**++[3
*]+*+[3 *]+*";                                         -232-
300 PRINT"{CYN}[40 ']";                                -118-
310 PRINT"{CYN}[40 ']";                                -119-
320 PRINT"{CYN}[40 ']";                                -120-
330 PRINT"{NRJ}(([5 )]()()()[6 (][6 )]()()()([3 )](()
()";                                                   -153-
340 PRINT"{RON}{NRJ}[40 ESP]";                         -086-
350 PRINT"{RON}{NRJ}[39 ESP]{ROF}"                     -142-
360 PRINT"{WHT}PUNTUACION {VCL}0","{WHT}NIVEL {VCL}1"
,"{WHT}VIDAS {VCL}[3 _]{HOM}"                          -048-
370 POKE 56255,8:POKE 1983,160:RETURN                  -175-
610 POKE 53272,21:POKE 789,234                         -142-
620 FOR S=0TO58:READA:POKE49152+S,A:NEXT S             -107-
630 SYS49152                                           -189-
640 READA:IF A=999 THEN 670                            -053-
650 FOR N=0 TO 7:READ B:POKE 14336+N+(A*8),B:NEXT N
                                                       -198-
660 GOTO 640                                           -175-
670 FOR S=0 TO 7:POKE 15616+S,255:NEXT S               -092-
675 FOR S=0 TO 7:POKE 14592+S,0:NEXT S                 -247-
680 FOR A=0 TO 38:READ N:POKE 51249+A,N:NEXT A
                                                       -211-
700 FOR S=0 TO 8 STEP 2:READ A,B,C,D:POKE 49500+S,A:P
OKE 49501+S,B                                          -057-
710 POKE 49600+S,C:POKE 49601+S,D:NEXT                 -235-
800 FOR S=54272 TO 54296:POKE S,0:NEXT S:POKE 54296,1
5                                                      -137-
810 POKE 54277,9:POKE 54278,0:POKE 54275,7:POKE 54274
,245                                                   -174-
820 POKE 54284,1:POKE 54285,128                        -197-
830 POKE 54291,4:POKE 54292,18                         -147-
900 FOR S=0TO6:N$(S)="-SLUGY-":P(S)=1000+(1000*S):NEX
T S                                                    -234-
1000 POKE 53269,0:FOR S=0TO127:READA:POKE832+S,A:NEXT
 S                                                     -071-
1010 POKE 53276,1:POKE 53285,1:POKE 53286,7:POKE 2040
,13                                                    -085-
1100 FOR S=0 TO 58:READ N:POKE 50000+S,N:NEXT S
                                                       -031-
1200 FOR S=0 TO 143:READ A:POKE 50500+S,A:NEXT S
                                                       -054-
1210 GOSUB 60080                                       -130-
5000 POKE 53265,9:POKE 53287,8:P=0:V=0:NI=1:DI=40
                                                       -047-
5010 POKE 53280,10:POKE 53281,0                        -172-
5020 PRINT"{CLR}{ABJ}{GRN}[3 ESP]TU[3 ESP]MICRO[3 ESP
]COMMODORE[3 ESP]PRESENTA"                             -036-
5030 PRINT"{RED}[3 ESP]!2 *![3 ESP]!5 *![3 ESP]!9 *![
3 ESP]!8 *!{2 ABJ}"                                    -119-
5050 PRINT"{ACL} {RON}{WHT}[6 ESP]{ROF}{ACL}[2 ESP]{R
ON}{WHT}[2 ESP]{ROF}{ACL}[6 ESP]{RON}{WHT}[2 ESP]{ROF
}[2 ESP]{RON}[2 ESP]{ROF}{ACL}[2 ESP]{RON}{WHT}[6 ESP
]{ROF}{ACL}[2 ESP]{RON}{WHT}[2 ESP]{ROF}{ACL}[2 ESP]{
RON}{WHT}[2 ESP]{ROF}{ACL} ";                          -198-
5060 PRINT"";                                          -247-
5070 PRINT"{ACL} {RON}{WHT}[6 ESP]{ROF}{ACL}[2 ESP]{R
ON}{WHT}[2 ESP]{ROF}{ACL}[6 ESP]{RON}{WHT}[2 ESP]{ROF
}{ACL}[2 ESP]{RON}{WHT}[2 ESP]{ROF}{ACL}[2 ESP]{RON}{
WHT}[6 ESP]{ROF}{ACL}[2 ESP]{RON}{WHT}[2 ESP]{ROF}{AC
L}[2 ESP]{RON}{WHT} ";                                 -251-
5080 PRINT" {ROF}{ACL} ";                              -101-
5090 PRINT"{ACL} {RON}{WHT}[2 ESP]{ROF}{ACL}[6 ESP]{R
ON}{WHT}[2 ESP]{ROF}{ACL}[6 ESP]{RON}{WHT}[2 ESP]{ROF
}{ACL}[2 ESP]{RON}{WHT}[2 ESP]{ROF}{ACL}[2 ESP]{RON}{
WHT}[2 ESP]{ROF}{ACL}[2 ESP]{RON}{WHT}[2 ESP]{ROF}{AC
L}[2 ESP]{RON}{WHT}[2 ESP]{ROF}{ACL} ";                -233-
5100 PRINT" {RON}{WHT}[2 ESP]{ROF}{ACL} ";             -181-
```

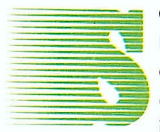
```
5110 PRINT"{ACL} {RON}{WHT}[2 ESP]{ROF}{ACL}[6 ESP]{R
ON}{WHT}[2 ESP]{ROF}{ACL}[6 ESP]{RON}{WHT}[2 ESP]{ROF
}{ACL}[2 ESP]{RON}{WHT}[2 ESP]{ROF}{ACL}[2 ESP]{RON}{
WHT}[2 ESP]{ROF}{ACL}[6 ESP]{RON}{WHT}[2 ESP]{ROF}{AC
L}[2 ESP]{RON}{WHT} ";                            -246-
5120 PRINT" {ROF}{ACL} ";                         -096-
5130 PRINT"{ACL} {RON}{WHT}[6 ESP]{ROF}{ACL}[2 ESP]{R
ON}{WHT}[2 ESP]{ROF}{ACL}[6 ESP]{RON}{WHT}[2 ESP]{ROF
}{ACL}[2 ESP]{RON}{WHT}[2 ESP]{ROF}{ACL}[2 ESP]{RON}{
WHT}[2 ESP]{ROF}{ACL}[6 ESP]{WHT}<*>{RON} <*>!\! {ROF
}!\!{ACL} ";                                      -177-
5140 PRINT"";                                     -246-
5150 PRINT"{ACL} {RON}{WHT}[6 ESP]{ROF} {ACL} {RON}{W
HT}[2 ESP]{ROF}{ACL}[6 ESP]{RON}{WHT}[2 ESP]{ROF}{ACL
}[2 ESP]{RON}{WHT}[2 ESP]{ROF}{ACL}[2 ESP]{RON}{WHT}[
2 ESP]{ROF}{ACL} {RON}{WHT}[3 ESP]{ROF}{ACL}[3 ESP]{W
HT}<*>{RON}[2 ESP]{ROF}!\!";                      -212-
5160 PRINT"{ACL}[2 ESP]";                         -210-
5170 PRINT"{ACL}[5 ESP]{RON}{WHT}[2 ESP]{ROF}{ACL}[2
ESP]{RON}{WHT}[2 ESP]{ROF}{ACL}[6 ESP]{RON}{WHT}[2 ES
P]{ROF}{ACL}[2 ESP]{RON}{WHT}[2 ESP]{ROF} {ACL} {RON}
{WHT}[2 ESP]{ROF}{ACL}[2 ESP]{RON}{WHT}[2 ESP]{ROF}{A
CL}[4 ESP]{RON}{WHT}[2 ESP]{ROF}{ACL} ";          -040-
5180 PRINT"[2 ESP]";                              -058-
5190 PRINT"{ACL}[5 ESP]{RON}{WHT}[2 ESP]{ROF}{ACL}[2
ESP]{RON}{WHT}[2 ESP]{ROF} {ACL}[5 ESP]{RON}{WHT}[2 E
SP]{ROF}{ACL}[2 ESP]{RON}{WHT}[2 ESP]{ROF}{ACL}[2 ESP
]{RON}{WHT}[2 ESP]{ROF}{ACL}[2 ESP]{RON}{WHT}[2 ESP]{
ROF}{ACL}[4 ESP]{RON}{WHT}[2 ESP]{ROF}{ACL} ";
                                                  -042-
5200 PRINT"[2 ESP]";                              -051-
5210 PRINT"{ACL} {RON}{WHT}[6 ESP]{ROF}{ACL}[2 ESP]{R
ON}{WHT}[6 ESP]{ROF}{ACL}[2 ESP]{RON}{WHT}[6 ESP]{ROF
}{ACL}[2 ESP]{RON}{WHT}[6 ESP]{ROF}{ACL}[4 ESP]{RON}{
WHT}[2 ESP]{ROF}{ACL}[3 ESP]";                    -221-
5220 PRINT"{ACL} {RON}{WHT}[6 ESP]{ROF}{ACL} {WHT} {R
ON}[6 ESP]{ROF}{ACL}[2 ESP]{RON}{WHT}[6 ESP]{ROF}{ACL
}[2 ESP]{RON}{WHT}[6 ESP]{ROF}{ACL}[4 ESP]{RON}{WHT}[
2 ESP]{ROF}{ACL}{8 ABJ}"                          -203-
5320 PRINT"{WHT}[2 ESP]PROGRAMADO POR:[2 ESP]{YEL}PAB
LO GARCIA MOLINA"                                 -042-
5330 POKE 1024,105:POKE 55296,10:POKE 1063,95:POKE553
35,10                                             -204-
5340 POKE 1984,223:POKE 56256,10:POKE 2023,233:POKE 5
6295,10                                           -039-
5350 FOR N1=1TO38:POKE55536+N1,4:NEXT N1          -130-
5360 FOR N1=1TO38:POKE55576+N1,5:NEXT N1          -136-
5370 FOR N1=1TO38:POKE55616+N1,6:NEXT N1          -133-
5380 FOR N1=1TO38:POKE55656+N1,7:NEXT N1          -139-
5390 FOR N1=1TO38:POKE55696+N1,8:NEXT N1          -145-
5400 FOR N1=1TO38:POKE55736+N1,9:NEXT N1          -133-
5410 FOR N1=1TO38:POKE55776+N1,10:NEXT N1         -178-
5420 FOR N1=1TO38:POKE55816+N1,11:NEXT N1         -175-
5430 FOR N1=1TO38:POKE55856+N1,14:NEXT N1         -183-
5440 POKE 53248,172:POKE 53249,188                -101-
5450 POKE 53269,1                                 -163-
5460 POKE 53265,27                                -216-
5470 FOR A=0 TO 12:IF PEEK(56320)=111 THEN 10000
                                                  -122-
5475 NEXT A                                       -149-
5480 IF PEEK(2040)=13 THEN POKE 2040,14:GOTO 5470
                                                  -245-
5490 POKE 2040,13:GOTO 5470                       -250-
10000 POKE 53280,8:POKE 53265,9:GOSUB 100:POKE 53269,
1:X=0:Y=0:NP=0                                    -087-
10005 FOR S=0TO3:POKE 53000+S,0:NEXT S            -250-
10007 FOR S=0TO2:POKE 53004+S,40:NEXT S           -051-
10010 POKE 53272,30:POKE 53248,168:POKE 53249,69
                                                  -120-
10020 POKE 2010,48+NI:POKE 251,0:POKE 253,0:POKE 2,0:
POKE 789,201                                      -130-
10030 FOR S=0TOV:POKE 2019+S,32:NEXT S:POKE 2040,13
                                                  -003-
10040 PRINT"{3 ABJ}{CYN}":POKE 53287,6            -172-
10045 FOR S=1 TO 4:X(S)=INT(RND(1)*30)-15:NEXT S
                                                  -026-
10050 PRINTTAB(X(1)+19)"{ABJ} {ABJ}{IZQ} "        -125-
10060 PRINTTAB(X(2)+19)" {ABJ}{IZQ} "             -110-
10070 PRINTTAB(X(3)+19)"{ABJ} {ABJ}{IZQ} "        -129-
10080 PRINTTAB(X(4)+19)" {ABJ}{IZQ} {VCL}{5 ABJ}"
                                                  -096-
11000 POKE 53265,27:X(0)=0                        -123-
11010 PRINTTAB(10)P"{ARB}"                        -078-
11990 X1=0:Y1=0                                   -075-
12000 J=PEEK(56320):IF PEEK(2)=1 THEN 25000       -117-
12005 IF INT(RND(1)*DI)=2 THEN GOSUB 39000        -154-
12010 IF J=119 THEN IF X<20 THEN X1=1:GOTO 12070
                                                  -236-
12020 IF J=123 THEN IF X>-19 THEN X1=-1:GOTO 12070
                                                  -076-
12030 IF J=126 THEN IF Y>1 THEN Y1=-1:GOTO 12070
                                                  -236-
12040 IF J=125 THEN IF Y<15 THEN Y1=1:GOTO 12070
                                                  -242-
12045 IF J=111 THEN 20000                         -070-
12050 GOTO 12000                                  -100-
12070 PO=1243+(40*(Y+Y1))+X+X1                    -096-
12075 IF PEEK(PO)=39 THEN 21000                   -238-
12077 IF PEEK(PO)=32 THEN 12080                   -241-
12079 GOTO 11990                                  -128-
12080 POKE PO,160:X=X+X1:Y=Y+Y1:GOTO 11990        -028-
13000 GOTO 13000                                  -097-
20000 SYS 50500                                   -043-
20010 X=X(PEEK(252))                              -096-
20020 Y=1+((PEEK(252)-1)*3)                       -114-
20040 IFPEEK(252)=0THENX=0:Y=0:POKE1323+X(1),32:POKE1
363+X(1),32:GOTO20060                             -020-
20045 POKE 1283+(40*Y)+X,32:POKE 1323+(40*Y)+X,32
                                                  -120-
20046 IF PEEK(252)=4 THEN 20060                   -178-
20047 POKE 1403+(40*Y)+X(PEEK(252)+1),32          -103-
20049 POKE 1443+(40*Y)+X(PEEK(252)+1),32          -109-
20060 FOR S=250 TO 0 STEP -50                     -103-
20070 POKE 54272,S:POKE 54273,S:POKE 54276,17
                                                  -029-
20080 NEXT S                                      -204-
20090 POKE 54276,16:IF PEEK(252)=0 THEN 11990
                                                  -221-
20100 GOTO 12070                                  -102-
21000 POKE 54272,96:POKE 54273,67:POKE 54276,65:POKE
54276,64                                          -119-
21010 NP=NP+1:P=P+10:PRINTTAB(10)P"{ARB}"         -001-
21020 IF NP<280 THEN 12080                        -162-
21030 POKE 789,234:NI=NI+1:DI=DI-5:GOTO 10000
                                                  -074-
25000 POKE 789,234:SYS 50000:POKE 53280,PEEK(56216):F
OR S=0 TO 8 STEP 2                                -028-
25020 POKE 54273,PEEK(49500+S):POKE 54273,PEEK(49501+
S)                                                -064-
25030 POKE 54280,PEEK(49600+S):POKE 54279,PEEK(49601+
S)                                                -071-
25040 POKE 54276,65:POKE 54283,17                 -044-
25050 POKE 54276,64:POKE 54283,16:NEXTS           -247-
25070 V=V+1:IF V<3 THEN 10000                     -113-
25080 FOR S=0TO2:POKE 54272,120:POKE 54273,10:POKE 54
276,65                                            -074-
25090 FOR N=0TO250STEP25:POKE54274,N:POKE 54275,N:NEX
T N                                               -153-
25100 POKE 54276,64                               -005-
25110 POKE 54275,7:POKE 54274,245                 -040-
25120 NEXT S:GOTO 40000                           -083-
35010 GOSUB 100:FOR S=0 TO H:SYS 50000:NEXT S:H=H+1:G
OTO 35000                                         -158-
39000 A=INT(RND(1)*7):POKE 253,PEEK(253)OR(2^A):RETUR
N                                                 -217-
39999 END                                         -014-
40000 POKE 53280,14:POKE 53265,9:POKE 53272,21:IF P<=
P(0) THEN 40220                                   -238-
40005 N$="":POKE 789,234:POKE 53281,3:X=18:X1=4
                                                  -089-
40010 A$="[7 ESP]A B C D E F G H I J K L M N O P Q R
S T U V W X Y Z !*!"                              -186-
40015 A$=A$+"[6 ESP]":B$="{RED}<2>!7 *!<X>{BLU}[9 ESP
]"                                                -118-
40020 PRINT"{BLK}{CLR}{3 ABJ}"TAB(7)"TU HABILIDAD[2 E
SP]PARA GUIAR EL"                                 -200-
40030 PRINTTAB(7)"{ABJ}APENDICE[2 ESP]NASAL DE SLUGY
TE"                                               -086-
40040 PRINTTAB(7)"{ABJ}INCLUYE ENTRE LOS 7 MEJORES:"
                                                  -228-
40050 POKE 53248,168:POKE 53249,140:POKE 2040,13
                                                  -105-
40060 PRINTTAB(12)"{5 ABJ}{NRJ}!U!!13 *!!I!"
                                                  -170-
40070 PRINTTAB(12)"!B!{BLU}A B C D E F G{NRJ}!B!"
                                                  -055-
40080 PRINTTAB(12)"!J!!5 *!!I! !U!!5 *!!K!"       -075-
40090 PRINT"{5 ABJ}"TAB(X)B$:POKE 53265,27        -076-
40100 J=PEEK(56320):IF J=119 THEN IF X1<27 THEN X1=X1
+1:GOTO 40200                                     -012-
40110 IF J=123 THEN IF X1>1 THEN X1=X1-1:GOTO 40200
                                                  -157-
40120 IF J=111 THEN 40140                         -072-
40130 GOTO 40100                                  -102-
40140 PRINT"{HOM}{14 ABJ}":W$=CHR$(64+X1):IF X1=27 TH
EN W$=" "                                         -240-
40150 FOR S=1 TO 4:POKE2040,14:FOR N=1TO100:NEXTN
                                                  -177-
40155 POKE 54287,10:POKE 54286,4:POKE 54290,129:POKE
54290,128                                         -160-
40160 POKE 2040,13:FOR N=1TO100:NEXTN             -184-
40165 POKE 54287,128:POKE 54286,67:POKE 54290,129:POK
E 54290,128:NEXT S                                -000-
40170 FOR N=1TO6:PRINTTAB(19)W$:PRINTTAB(19)"{ARB} "
                                                  -218-
40175 POKE 54287,N*20:POKE 54286,N*20:POKE 54290,17
                                                  -057-
40178 FOR S=1TO10:NEXTS:NEXT N:POKE 54290,16
                                                  -195-
40180 X=X-1:N$=N$+W$:PRINT"{ARB}"TAB(X+1)N$:PRINTTAB(
X)B$:IF X>11 THEN 40100                           -069-
40185 GOTO  40220                                 -147-
40190 GOTO 40100                                  -108-
40200 PRINT"{HOM}{14 ABJ}"TAB(13)MID$(A$,X1*2,13)
                                                  -201-
40205 POKE 54287,39:POKE 54286,35:POKE 54290,17
                                                  -083-
40210 FOR S=1TO10:NEXTS:POKE54290,16:GOTO 40100
                                                  -055-
40220 POKE 53280,5:POKE 53265,9:POKE 53281,2:POKE 532
69,0                                              -149-
40230 PRINT"{CLR}{WHT}{2 ABJ}"TAB(15)"GAME OVER"
```

```
                                                         -236-
40240 PRINTTAB(15)"(CYN)!9 *!"                           -216-
40250 FOR S=6 TO 0 STEP-1:IF P>P(S) THEN 40265
                                                         -144-
40260 NEXT S:GOTO 40270                                  -094-
40265 FOR N=0 TO S-1:P(N)=P(N+1):N$(N)=N$(N+1):NEXT N
                                                         -004-
40267 P(S)=P:N$(S)=N$                                    -163-
40270 PRINT"(ABJ)"                                       -255-
40280 FOR S=6 TO 0 STEP  -1                              -248-
40290 P$="[3 ESP]"+STR$(P(S)):PRINT"(ABJ)(BLK)",RIGHT
$(P$,5),"(YEL)"N$(S)                                     -197-
40300 NEXT S:POKE 53265,27:A=1                           -213-
40310 FOR S=0TO8:POKE 55391+S,A:NEXT S                   -033-
40315 IF PEEK(56320)=111 THEN 5000                       -070-
40320 IF A=1 THEN A=0:GOTO 40310                         -031-
40330 A=1:GOTO 40310                                     -084-
41000 REM *** CARACTERES/ LM ***                         -026-
41010 DATA 234,234,234,234,234,234,173,14,220,41,254,
141,14,220,165,1                                         -106-
41020 DATA 41,251,133,1,162,0,189,0,208,157,0,56,232,
224,0,208                                                -016-
41030 DATA 245,189,0,209,157,0,57,232,224,0,208,245,1
65,1,9,4                                                 -245-
41040 DATA 133,1,173,14,220,9,1,141,14,220,96
                                                         -207-
41500 REM *** CARACTERES ***                             -055-
41510 DATA35,0,0,62,255,255,126,54,51                    -101-
41520 DATA36,34,20,124,254,254,252,96,112                -049-
41530 DATA37,68,40,62,127,127,63,6,14                    -115-
41540 DATA38,0,0,124,255,255,126,108,204                 -250-
41550 DATA39,0,0,24,60,60,24,0,0                         -093-
41560 DATA40,1,139,223,255,255,255,255,255               -111-
41570 DATA41,128,209,251,255,255,255,255,255
                                                         -218-
41580 DATA42,255,255,255,255,255,251,209,128
                                                         -220-
41590 DATA43,255,255,255,255,255,223,139,1               -117-
41600 DATA44,0,3,63,15,63,127,127,127                    -102-
41610 DATA45,127,127,127,63,15,63,3,0                    -104-
41620 DATA46,0,192,252,240,252,254,254,254               -102-
41630 DATA47,254,254,254,252,240,252,192,0               -104-
41640 DATA 999                                           -004-
42000 REM  *** MOVIMIENTO SLUGY ***                      -049-
42010 DATA 166,251,224,15,240,6,232,134,251,76,49,234
,162,0,134,251                                           -023-
42020 DATA 173,248,7,201,13,208,8,169,14,141,248,7,76
,49,234,169                                              -162-
42030 DATA 13,141,248,7,76,49,234                        -157-
44500 REM *** SONIDO ****                                -083-
44510 DATA 40,200,81,161,43,52,86,105                    -085-
44520 DATA 38,126,76,252,34,75,68,149                    -118-
44530 DATA 28,214,57,172                                 -229-
45000 REM *** DOS SPRITES ***                            -137-
45005 DATA 0,40,0,128,170,2,130,170,130,162,170,138,1
69,170,106,165                                           -013-
45010 DATA 105,90,166,105,154,37,105,88,41,170,104,10
,170,160,10,190                                          -065-
45020 DATA 160,2,190,128,8,190,32,42,190,168,42,190,1
68,42,190,168                                            -251-
45030 DATA 42,190,168,40,190,40,168,190,42,168,255,42
,43,255,232,0                                            -245-
45040 DATA 0,40,0,128,170,2,130,170,130,162,170,138,1
69,170,106,165                                           -012-
45050 DATA 105,90,167,105,218,37,105,88,41,170,104,10
,170,160,10,190                                          -071-
45060 DATA 160,2,190,128,8,255,32,42,255,168,42,255,1
68,43,255,232                                            -000-
45070 DATA 43,255,232,42,255,168,170,60,170,168,60,42
,168,60,42,0                                             -197-
50000 REM *** COLORES ***                                -108-
50010 DATA 162,0,189,0,216,24,105,1,157,0,216,232,224
,0,208,242                                               -060-
50020 DATA 189,0,217,24,105,1,157,0,217,232,224,0,208
,242,189,0                                               -072-
50030 DATA 218,24,105,1,157,0,218,232,224,0,208,242,1
89,0,219,24                                              -123-
50040 DATA 105,1,157,0,219,232,224,231,208,242,96
                                                         -161-
50500 REM *** LENGUA / LM ***                            -030-
50510 DATA 160,4,169,0,133,252,162,0,189,208,6,201,16
0,208,7,169                                              -138-
50520 DATA 32,157,208,6,132,252,232,224,120,208,237,1
65,252,201,0,240                                         -113-
50530 DATA 1,96,136,162,0,189,88,6,201,160,208,7,169,
32,157,88                                                -067-
50540 DATA 6,132,252,232,224,40,208,237,165,252,201,0
,240,1,96,136                                            -226-
50550 DATA 162,0,189,224,5,201,160,208,7,169,32,157,2
24,5,132,252                                             -190-
50560 DATA 232,224,40,208,237,165,252,201,0,240,1,96,
136,162,0,189                                            -234-
50570 DATA 104,5,201,160,208,7,169,32,157,104,5,132,2
52,232,224,40                                            -226-
50580 DATA 208,237,165,252,201,0,240,1,96,136,162,0,1
89,240,4,201                                             -182-
50590 DATA 160,208,7,169,32,157,240,4,132,252,232,224
,40,208,237,96                                           -044-
60080 FOR S=0TO95:READA:POKE 51505+S,A:NEXT S
                                                         -237-
60090 FOR S=0 TO 3:POKE S+53000,0:NEXT S                 -067-
60100 FOR S=0 TO 2:POKE 53004+S,40:NEXT S                -114-
60110 FOR S=0TO86:READA:FOR N=0 TO 3                     -153-
60120 POKE 51700+S+(N*100),A:NEXT N,S                    -059-
60130 FOR N=1TO3:A=N*100:POKE 51701+A,8+N:POKE 51784+
A,8+N                                                    -031-
60140 POKE 51708+A,108+(2*N):POKE 51714+A,109+(2*N)
                                                         -169-
60150 POKE 51731+A,108+(2*N):POKE 51745+A,109+(2*N)
                                                         -170-
60160 POKE 51772+A,108+(2*N):POKE 51778+A,109+(2*N)
                                                         -182-
60170 POKE 51768+A,2^N:NEXT N                            -213-
60180 POKE 51811,222:POKE 51812,5:POKE 51817,223:POKE
 51818,5                                                 -094-
60190 POKE 51820,224:POKE 51821,5:POKE 51834,225:POKE
 51835,5                                                 -097-
60200 POKE 51850,224:POKE 51851,5:POKE 51855,225:POKE
 51856,5                                                 -101-
60210 POKE 51875,6:POKE 51876,6:POKE 51881,7:POKE 518
82,6                                                     -176-
60220 POKE 51911,206:POKE 51912,6:POKE 51917,207:POKE
 51918,6                                                 -099-
60230 POKE 51920,208:POKE 51921,6:POKE 51934,209:POKE
 51935,6                                                 -102-
60240 POKE 51950,208:POKE 51951,6:POKE 51955,209:POKE
 51956,6                                                 -115-
60250 POKE 51975,246:POKE 51976,6:POKE 51981,247:POKE
 51982,6                                                 -132-
60260 POKE 52011,30:POKE 52012,7:POKE 52017,31:POKE 5
2018,7                                                   -223-
60270 POKE 52020,32:POKE 52021,7:POKE 52034,33:POKE 5
2035,7                                                   -226-
60280 POKE 52050,32:POKE 52051,7:POKE 52055,33:POKE 5
2056,7                                                   -239-
60290 POKE 52075,70:POKE 52076,7:POKE 52081,71:POKE 5
2082,7                                                   -000-
60500 REM *** DIRECCIONES/ LM ***                        -107-
60510 DATA 166,254,232,224,30,240,5,134,254,76,49,200
,162,0,134,254                                           -026-
60520 DATA 165,253,41,1,201,1,208,3,32,244,201,165,25
3,41,2,201                                               -058-
60530 DATA 2,208,3,32,88,202,165,253,41,4,201,4,208,3
,32,188                                                  -189-
60540 DATA 202,165,253,41,8,201,8,208,3,32,32,203,165
,253,41,16                                               -077-
60550 DATA 201,16,208,3,32,132,203,165,253,41,32,201,
32,208,3,32                                              -111-
60560 DATA 232,203,165,253,41,64,201,64,208,3,32,76,2
04,76,49,200                                             -190-
61000 REM *** DERECHA/ LM ***                            -043-
61010 DATA 174,8,207,224,0,240,12,173,108,207,157,238
,4,173,109,207                                           -029-
61020 DATA 157,239,4,189,240,4,201,160,208,4,160,1,13
2,2,141,108                                              -121-
61030 DATA 207,189,241,4,201,160,208,4,160,1,132,2,14
1,109,207,169                                            -223-
61040 DATA 35,157,240,4,169,36,157,241,4,232,232,224,
40,208,20,162                                            -232-
61050 DATA 0,165,253,73,1,133,253,173,108,207,141,22,
5,173,109,207                                            -230-
61060 DATA 141,23,5,142,8,207,96                         -102-
61100 REM *** IZQUIERDA/ LM ***                          -238-
61110 DATA 174,12,207,224,40,240,12,173,116,207,157,1
04,5,173,117,207                                         -116-
61120 DATA 157,105,5,189,102,5,201,160,208,4,160,1,13
2,2,141,116                                              -112-
61130 DATA 207,189,103,5,201,160,208,4,160,1,132,2,14
1,117,207,169                                            -221-
61140 DATA 37,157,102,5,169,38,157,103,5,202,202,224,
0,208,20,162                                             -175-
61150 DATA 40,165,253,73,16,133,253,173,116,207,141,1
04,5,173,117,207                                         -128-
61160 DATA 141,105,5,142,12,207,96                       -195-
61170 FOR S=0 TO 86:READ A:FOR N=0TO2                    -191-
61180 N1=N*100:POKE 52100+S+N1,A:NEXT N,S                -097-
61190 FOR S=1TO2:N=S*100:POKE 52168+N,2^(S+4)
                                                         -088-
61200 POKE 52101+N,12+S:POKE 52184+N,12+S                -250-
61210 POKE 52108+N,116+(S*2):POKE 52114+N,117+(S*2)
                                                         -192-
61220 POKE 52131+N,116+(S*2):POKE 52145+N,117+(S*2)
                                                         -193-
61230 POKE 52172+N,116+(S*2):POKE 52178+N,117+(S*2):N
EXT S                                                    -185-
61240 POKE 52211,88:POKE 52212,6:POKE 52217,89:POKE 5
2218,6                                                   -254-
61250 POKE 52220,86:POKE 52221,6:POKE 52234,87:POKE 5
2235,6                                                   -249-
61260 POKE 52250,86:POKE 52251,6:POKE 52255,87:POKE 5
2256,6                                                   -006-
61270 POKE 52275,88:POKE 52276,6:POKE 52281,89:POKE 5
2282,6                                                   -023-
61280 POKE 52311,248:POKE 52312,6:POKE 52317,249:POKE
 52318,6                                                 -098-
61290 POKE 52320,246:POKE 52321,6:POKE 52334,247:POKE
 52335,6                                                 -093-
61300 POKE 52350,246:POKE 52351,6:POKE 52355,247:POKE
 52356,6                                                 -097-
61310 POKE 52375,248:POKE 52376,6:POKE 52381,249:POKE
 52382,6:RETURN                                          -140-

READY.                                                   -173-
```

El perfil del Amiga

Destinado a ocupar el vacío existente en Commodore, en el segmento medio-alto, el Amiga ha dado pie a ríos de tinta en su país de origen: U.S.A. Pero veamos sus peculiaridades.



Se veía venir. Con el abaratamiento progresivo de los microprocesadores y de los chips de memoria, ya nada impedía ofrecer al usuario del ordenador personal, un producto «como Dios manda». Este es el caso del Amiga, que con sus 256 Kbytes de memoria base y un microprocesador Motorola MC68000, abre todo un campo de posibilidades en la gama de los personales: potencia y velocidad.

Pero, antes de entrar en materia, ¿qué es el Amiga? El Commodore Amiga es un ordenador no desarrollado íntegramente por Commodore, por lo menos en su base inicial; tanto es así, que la firma norteamericana no quiere que se le denomine Commodore Amiga sino Amiga by Commodore, e incluso, junto con el conocido logotipo de Commodore figura el específico del Amiga. El conjunto consta de cuatro partes bien diferenciadas: la unidad central, el teclado, el monitor color RGB y el ratón.

La unidad central es la que contiene el quiz de la cuestión (para más información ver «Amiga un interior con miga» en el número 3-4 de nuestra publicación); formada por una caja independiente del teclado que contiene el microprocesador (impresionante por su tamaño), la memoria y toda la circuitería (¿o habría que decir la «chipería»?), contiene también todos los «slots» (conectores) de comunicación, entre los que cabe destacar la salida RS-232 y el port paralelo centronics para conectar casi cualquier impresora. Esta unidad está sobreelevada, con lo cual podemos guardar el teclado debajo de ella, cuando no se esté usando.

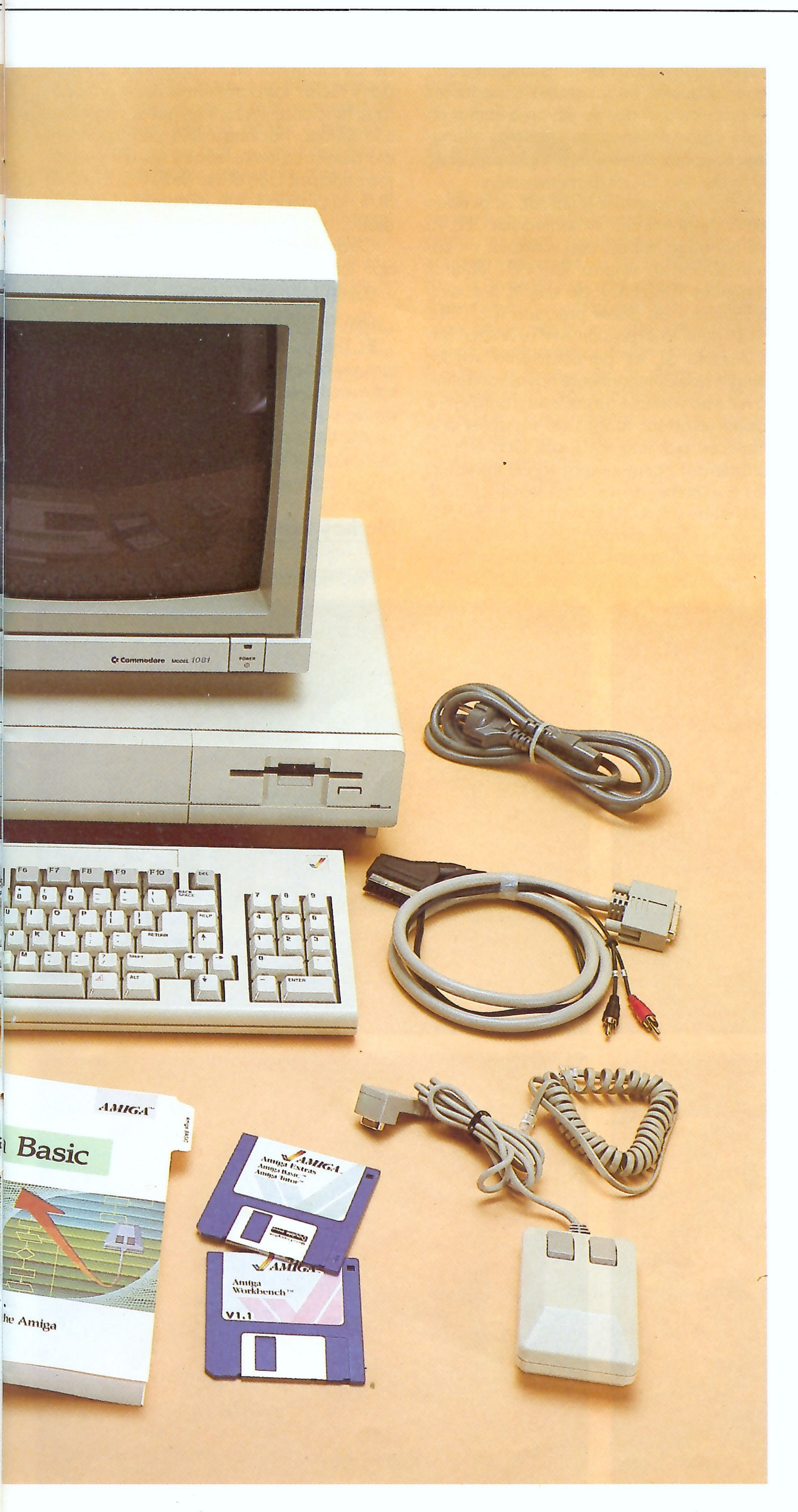
El teclado, que se conecta a la unidad como otro periférico más, dispone de dos patas en el fondo, para elevarlo cuando se utilice, dándole así una inclinación más cómoda y natural. La presión que hay que ejercer sobre las teclas es mucho más suave que en el C64 (menor recorrido), y la calidad del conjunto es magnífica; dispone de 10 teclas de función y de keypad numérico; este último, se queda un poco corto al no disponer en el conjunto de tecla de punto, ni de ningún operador aritmético, excepto el del signo menos.

En cuanto al monitor en color, creemos que no se puede pedir más, al tratarse de uno del tipo RGB (algo así como el Rolls Roice de los monitores), la calidad del color, así como la nitidez y contraste, es superior, dando realce a la delicada e impresionante gama de colores con los que el Amiga nos puede obsequiar.

El cuarto componente de hardware es el ratón. Cuando se trata de explicar lo que es un ratón, se encuentran pocos usos prácticos; pues bien, en el Amiga el ratón es poco menos que imprescindible. Con él, manejamos un icono (una flechita), que será nuestro «dedo electrónico», gracias al cual no sólo efectuamos selecciones, sino que además, indicamos acciones, lo cual nos permite trabajar a pleno rendimiento de una forma completamente independiente al teclado.

Por supuesto, no todas las tareas son accesibles sin el concurso del teclado, pero, por ejemplo, para todo lo que sea gestión de discos (copiado, organización de directorios, duplicado de ficheros, creación de ventanas de salida para discos, posicionamiento de iconos en pantalla, conexión de impresoras específicas, selección de número de columnas, y un largo etc.) o procesos intermedios, no necesitaremos el teclado en absoluto.





Ventanas, iconos y menús

La filosofía del Amiga, puede muy bien quedar resumida en las tres palabras que encabezan este párrafo: ventanas, iconos y menús. Vamos a ver en qué consiste cada uno de estos elementos.

En primer lugar, las ventanas son pantallas totalmente INDEPENDIENTES una de otra, que se pueden abrir y cerrar (lógica aplastante), cambiar de tamaño y desplazar de un lugar a otro de la pantalla (nos referimos ahora a la pantalla física del monitor) o cambiarlas de plano delantero a plano trasero (pensemos en las ventanas abiertas como si fuesen un paquete de fichas que pasamos de adelante hacia atrás o viceversa). De esta manera, podemos tener a la vez tantas ventanas abiertas como nos permita la memoria (no, no..., la del ordenador), y por supuesto, redirigir la salida de un programa hacia cualquiera que deseemos.

Para ver lo que es un icono, y su manejo por parte del Amiga, apaguemos el ordenador y volvamos a encenderlo; al cabo de unos segundos, un dibujo alusivo nos indicará que introduzcamos el disco Kickstart en el drive. Tras un breve lapso de tiempo, la misma imagen nos instará a introducir el disco Workbench; una vez cumplido este trámite, aparecerá el dibujito de un disco (icono de disco) con el rótulo «Workbench» debajo de él.

A continuación, extraemos el disco del drive e introducimos el etiquetado «Extras». Nuestro inteligente Amiga, rastrea periódicamente el drive en busca de nuevos discos, al encontrar ahora el «Extras», un nuevo icono de disco designado «Extras» aparecerá en la pantalla. Si queremos ver su contenido, desplazamos la flecha sobre el mismo y pulsamos dos veces el botón izquierdo del ratón, resultado: una ventana titulada «Extras», se abre con una serie de iconos en su interior; estos iconos pueden ser cajones (subdirectorios), herramientas, aplicaciones, ficheros, programas, etc., que a su vez se pueden seleccionar con el ratón para arrancar el programa o copiar el fichero en otro disco, guardarlo en un cajón...

Esta filosofía de los iconos, popularizada por el célebre Apple Mackintosh, es un intento de ponerse en un contacto con el usuario de una forma más conocida y amigable.

Los menús, son otra característica peculiar del amiga. Son del tipo «pull-down», algo así como «tira hacia abajo», y la verdad que la palabreja en inglés los define perfectamente, ya que al seleccionar el título de un menú, instantáneamente se desarrollan hacia abajo todas las posibilidades a

Programación estructurada

¿Quién dijo que el BASIC no es un lenguaje estructurado? Antes de nada, diremos que el BASIC puede ser un lenguaje tan estructurado como cualquier otro... si le damos las herramientas necesarias. BASIC de Amiga es una versión del popular BASIC de Microsoft denominado Amiga BASIC, en él, se dan todas las características necesarias para realizar una programación limpia e impecable, modular y estructurada. Aunque también se puede programar de forma más anárquica, con soluciones «al paso», típica de programadores noveles. Entre los ingredientes necesarios para programar perfectamente, vamos a destacar algunos de ellos.

El Amiga considera los números de línea como etiquetas, y son opcionales al igual que la pulsación de la tecla RETURN al editar (podríamos decir que tecleamos directamente en la memoria del programa). Este tratamiento de los números de línea, hace posible que la línea 100 PRINT, Puede ir después que la 200 PRINT. Es suficiente con poner etiquetas en las partes importantes del programa, o que se vayan a llamar en algún sitio, admitiéndose etiquetas alfanuméricas como: mueveratón, y por supuesto, saltos de tipo GOSUB mueveratón.

Todas las variables a utilizar, se pueden definir de antemano en sentencias DEF, ya que el Amiga trabaja con variables de cadena (DEFSTR), enteras (DEFINT), enteros ampliados (DEFLNG), de simple precisión (DEFSNG) y de doble precisión (DEFDBL); con lo cual, la expresión A="Tu Micro" (sin el símbolo $), es perfectamente válida, siempre que anteriormente hayamos definido A como variable de cadena. También es posible expresar rangos de variables que empiecen por una letra, así: DEFINT i—l, considerará como enteras todas las variables que comiencen por las letras i, j, k o l.

El uso de subprogramas, hará las delicias del buen programador, ya que se pueden mantener bibliotecas de subprogramas generales e incorporar los necesarios a nuestro programa principal. Estos se diferencian de las subrutinas en la capacidad de mandar y recibir los parámetros necesarios para su funcionamiento, y en que sus variables son únicas, haciendo una correspondencia biunívoca entre éstas y las del programa principal. Con el uso de EXIT SUB y END SUB, la claridad que se consigue es meridiana.

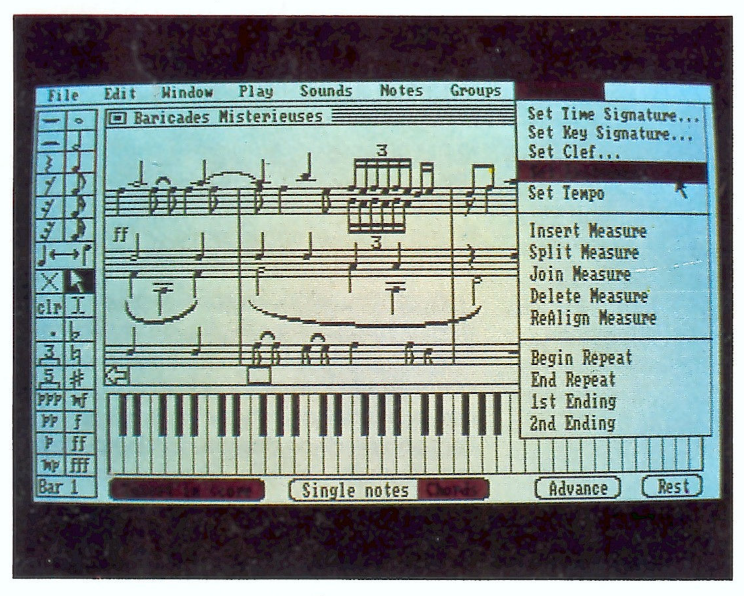
Otras sentencias típicas de la programación estructurada son:

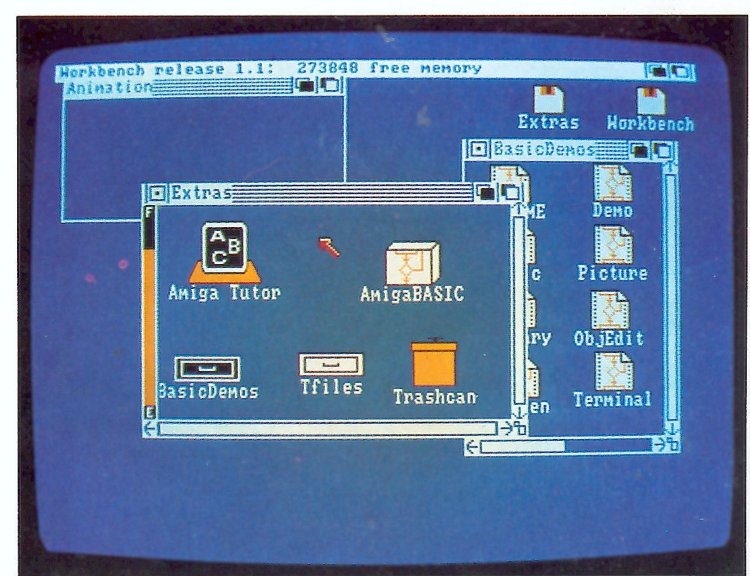
WHILE—WEND, que permite hacer bucles condicionales.

IF...THEN...ELSE..ELSEIF..END IF: la estructura de las comparaciones en el Amiga es estupenda; se pueden contemplar todo tipo de casos y además, es posible separar las condiciones de un IF en bloques de sentencias, gracias a ELSEIF y a END IF.

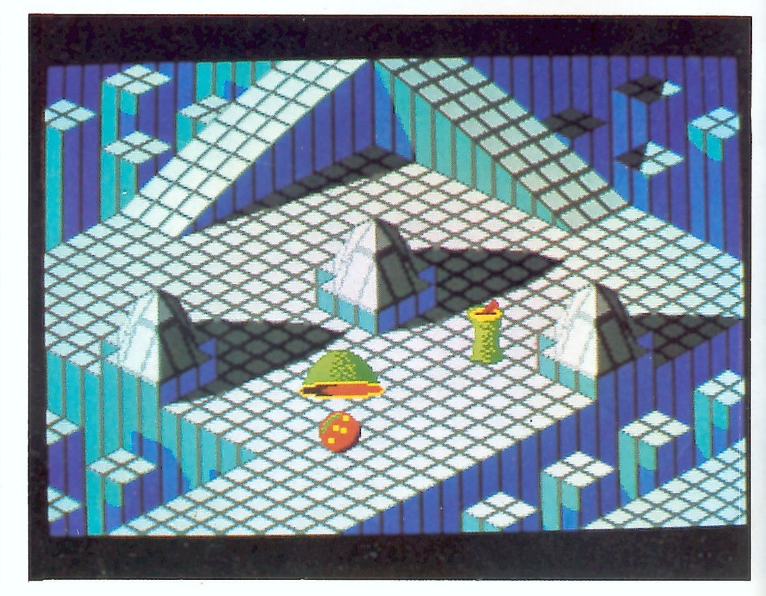
No hay bien que cien líneas dure

Pero no todo iba a ser bueno en el Ami-









ga, vamos ahora a desvelar los trapos sucios que los tiene (y algunos gordos).

En primer lugar, parece que con el equipo de deberían entregar más manuales de los que vienen (Amiga BASIC e Introducción al Amiga), ya que en estos dos libros, se hacen constantes referencias a otros dos (por lo menos), que son el Amiga DOS y el Kernel ROM. Así, nos encontramos, justo en las características más interesantes, con un «...para mayor información ver el capítulo... del Amiga DOS» ¿Dónde están estos manuales?

Por otra parte, llegan rumores desde U.S.A. que la versión del Workbench 1.1 (es de la única que disponemos), tiene una serie de fallos importantes, que se ponen de manifiesto justo cuando son menos esperados, y que pueden ser debidos a una excesiva rapidez en su confección. También sabemos que en las tierras del tío Sam, están preparando una nueva versión modificada y corregida (Workbench 1.2). Aunque no está claro qué fallos son achacables a la versión del Workbench, aquí enumeramos los que nosotros hemos encontrado (por desgracia para unas cuantas horas de trabajo).

A meditar con el Guru

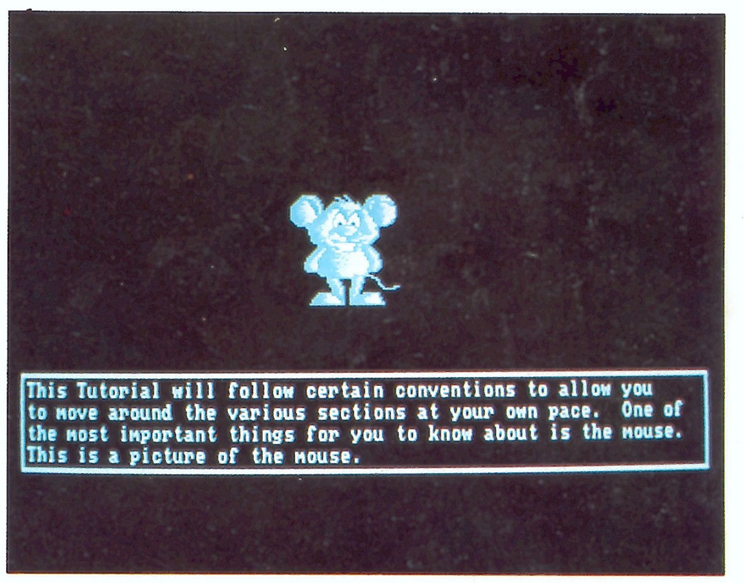
De vez en cuando, y ante fallos tan nímios como olvidarse un RETURN, o seleccionar un menú sin destino, el ordenador entra en una fase de guiños epilépticos que acaba, irremisiblemente, con la caída del sistema (y la nuestra por no haber grabado antes de...) y con un mensaje curiosísimo que puede ser algo así: Software failure... Guru Meditation (numerajo). Es decir, que nos mandan poco más o menos que a hacer meditación trascendental, mientras el sistema se «resetea» y nos vuelve a pedir el disco Workbench. Si no lo veo, no lo creo. Moraleja: haz siempre una grabación, para evitar la meditación.

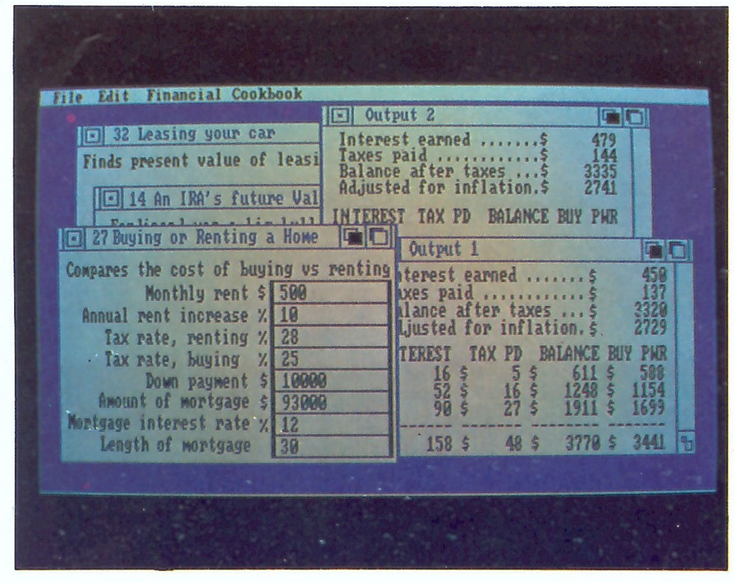
Al insertar caracteres en la ventana de listado, si mueves el cursor al final de la línea (que no se ve en la ventana por su longitud), ¡habrás perdido lo insertado!; además, en algunas ocasiones, al insertar algo en una línea y tener que efectuar un salto a la derecha (para mostrar el final de la línea), se mezclará parte de lo insertado con parte del final de la misma. Hay que asegurarse bien de que la corrección deseada se ha hecho, antes de continuar.

Por último, los mensajes de error son demasiado escuetos en algunas ocasiones. Se dispone solamente de 38 mensajes de error (sin contar los del disco), dado que los códigos de error del 39 al 49 ambos inclusive, no tienen mensaje asignado, apareciendo sin embargo, uno para todos ellos: UNPRINTABLE ERROR. Un poco extraño sí que es, porque ya que tienen un mensaje, podrían haber profundizado un poco más.

En definitiva, un equipo con unas capacidades gráficas, sonoras y de programación excepcionales, aunque flojo en algunos puntos, como puede ser la detección de errores por parte del sistema, aunque en su descargo hay que decir que esto último parece en vías de solución.











nuestro alcance.

Para seleccionar un menú, se utiliza el botón derecho del ratón, que precisamente se denomina botón de menú; al pulsarlo, aparecen en la parte superior de la ventana los rótulos correspondientes a los títulos de los menús disponibles. Aparte de esto, nada de particular parece suceder, pero (sin dejar de pulsar el botón de menú) acerquemos la flecha hasta tocar alguno de los títulos; inmediatamente el título se remarca en reverso y una «tira» se descuelga debajo del mismo, con todas las posibilidades que ese menú nos ofrece. Si deseamos optar por alguna, bajemos hacia ella sin soltar el botón todavía, a nuestro paso, las opciones se irán poniendo en reverso, y al llegar a la deseada, bastará con soltar el botón para que se «ejecute» nuestra elección.

Estas tres características del Amiga, están a nuestra completa disposición desde el BASIC, con sentencias como: MENU ON, ON MENU..., MENU, etc., podemos desarrollar programas conducidos a base de menús, donde el usuario podrá actuar casi exclusivamente con el ratón. Igualmente, disponemos de toda una serie de sentencias para el manejo de las ventanas (WINDOW..., WINDOW CLOSE, WINDOW OUTPUT, etc.) que nos permiten definir las ventanas de salida, con las características que nosotros deseemos (tamaño, posición, prioridad, etc.). En cuanto a los iconos, podremos diseñarlos a nuestro antojo con el programa Editor de Iconos, que acompaña al equipo.

ON todo GOSUB...

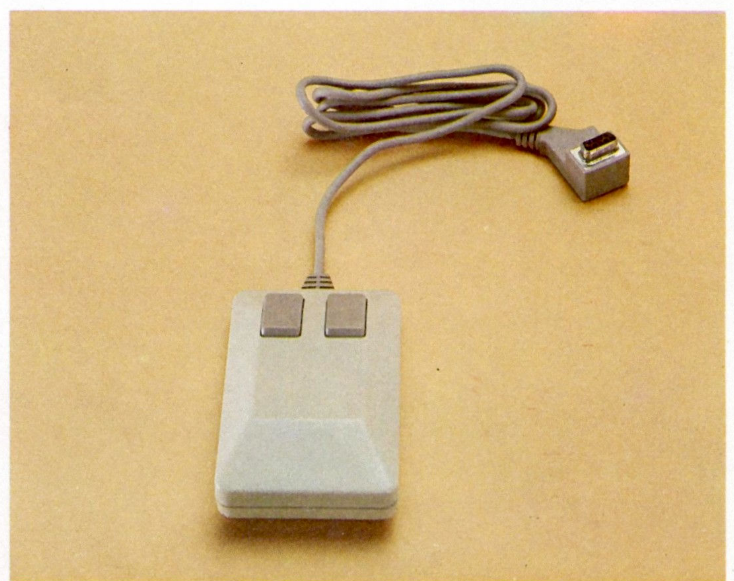
Una sentencia común en BASIC, es ON x GOTO (GOSUB)..., en el BASIC del Amiga, tenemos a nuestra disposición toda una serie de sucesos que podemos «atrapar» con sentencias como la anterior, lo cual hace posible un tipo de programación ideal para su utilización con usuarios no especializados en informática, a la vez que nos abre las puertas de un mundo fascinante: la multitarea o multiprogramación.

En efecto, con el uso de la sentencia ON TIMER, podemos redirigir el flujo del programa a distintas partes del mismo (multiprogramación), cada vez que pasa una cierta cantidad de tiempo. Los distintos sucesos que el Amiga puede detectar son: ON BREAK, cada vez que el usuario intenta detener el programa; ON COLLISION, al colisionar un objeto con otro o con un borde; ON ERROR, por producirse un error; ON MENU, cuando el usuario selecciona un menú; ON MOUSE, cada vez que se pulsa el botón izquierdo del ratón; y ON TIMER, al paso de un cierto número de segundos.









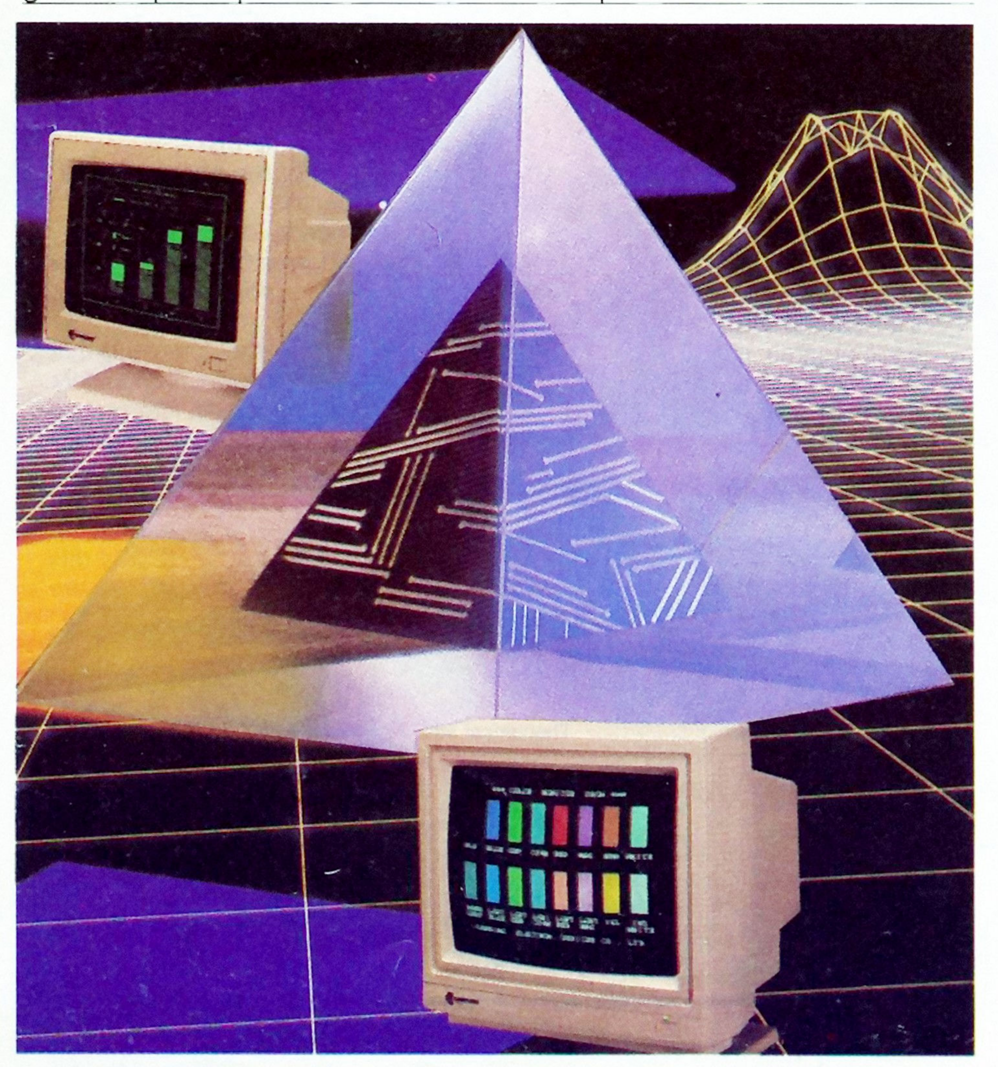
¡Un millón de colores!

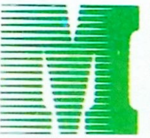
Una de las características más espectaculares del Amiga, es que podemos escoger nuestro propio juego de colores entre un total de 4096, pudiendo tener a la vez, como máximo, 32 de ellos. No obstante, al menos en teoría, la gama cromática del aparato parece ser de ¡un millón!

Para aclarar este aspecto, debemos ir a la forma de definir los colores con la sentencia PALETTE; ésta irá acompañada además del número del color a cambiar, de tres porcentajes expresados en tanto por uno (0-1), de los colores primarios, es decir, rojo, verde y azul. En teoría podemos dar 100 valores a cada color, lo que supondría 100x100x100=1000000 de combinaciones posibles. La limitación estriba en que el sistema «sólo» reconoce 16 incrementos para cada primario, con lo que el número de colores real queda en: 16x16x16=4096; aún así, los cambios que produce en un color una variación mínima de un primario, son difíciles de detectar para un no iniciado, con lo cual, es bastante razonable decir, que el Amiga puede representar prácticamente cualquier tonalidad que deseemos.

Gráficos tridimensionales: el espacio en tu Commodore

Representar el espacio es algo que el hombre ha hecho desde el principio de los tiempos, desde las pinturas rupestres a los clásicos renacentistas, pasando por la geometría y el dibujo técnico. Con los ordenadores, la magia del espacio puede ser encerrada en la pantalla de nuestro televisor.



uchos de vosotros habréis visto la película EL ULTIMO STARFIGHTER, o la ya clásica TRON. Y prácticamente todos os habréis sorprendido al ver sus imágenes, y más al saber que buena parte de ellas sólo han existido en la memoria de un ordenador, y nunca en la realidad. El campo de los gráficos generados por ordenador está en sus comienzos, pero ya recrea nuestra imaginación con mundos lejanos e inexistentes.

Pero no todo es entretenimiento en este campo de la informática. Pensad, por ejemplo, en los simuladores de vuelo, donde los pilotos se enfrentan una y otra vez a situaciones peligrosas sin arriesgar ni sus vidas ni sus aviones; y pronto veremos simuladores para automóviles, como el que actualmente desarrolla Mercedes-Benz, donde podremos realizar acrobacias o conducir por autopistas infestadas de tráfico durante horas, sin peligro para nosotros (ni para el profesor, que también pasa sus sustos...).

Pero cuando pensamos en todos estos campos, imaginamos ordenadores enormes (como el gigantesco CRAY-1, utilizado en la confección de EL ULTIMO STARFIGHTER), o especialistas en informática que pasan sus vidas programando para que aquello funcione.

Sin embargo, si nos conformamos con gráficos más sencillos, podemos programarlos nosotros mismos. Incluso un buen simulador de vuelo está dentro del alcance de nuestro Commodore (como el fabuloso JET, de Sublogic).

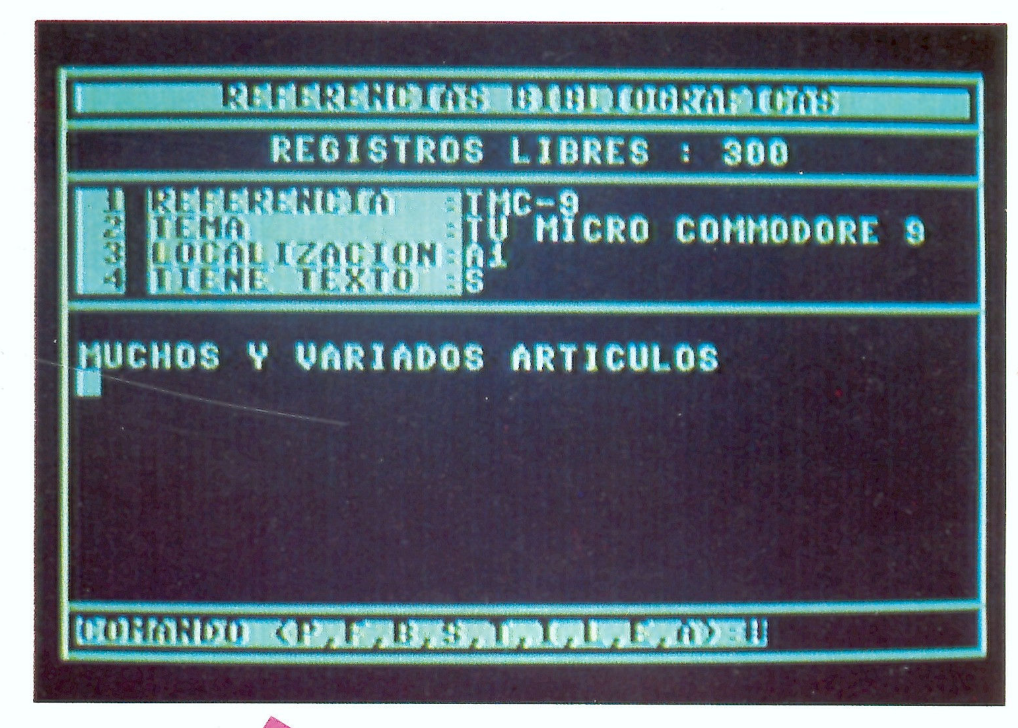
Vamos a ver las bases de la creación de gráficos tridimensionales. Por la amplitud del tema, nos centraremos en la representación de funciones de dos variables, aun-

Página faltante

Página faltante

que no queden palabras partidas.

Pulsando la tecla **F7** se entra en el modo «COMANDO» del editor, el cual dispone de las siguientes funciones: «P» sitúa el cursor al principio del texto. «F» sitúa el cursor al final del mismo. «I» inserta una línea en la posición del cursor. «L» borra la línea en la que se encuentra el cursor. «B» busca una palabra o texto. «S», una vez localizada una palabra, busca la siguiente. «E» escribe el texto en la impresora. Por último, «T» termina el modo editor grabando el texto y volviendo a la primera pregunta.



Análisis de inversiones

Encontrar la inversión más adecuada para el dinero es un problema de permanente actualidad. Sin embargo, y pese a la abundancia de ofertas existente en los mercados financieros, no siempre se tiene un conocimiento suficiente de las alternativas posibles y de la rentabilidad real de las mismas. La proliferación de cifras de tantos por ciento de interés, ventajas fiscales, y otros datos no comparables necesariamente entre sí, incrementa la confusión que puede haber sobre este tema. Los cálculos financieros, por otro lado, no resultan en ocasiones fáciles para un profano en la materia.

El programa ANALISIS DE INVERSIONES es un sencillo instrumento, que pretende ayudar a plantear las inversiones persona-

HEADER PARA DISCOS PCW

les y a controlar la rentabilidad de las mismas. Un uso adecuado de este programa permitirá seleccionar en cada momento la inversión que proporcione mayor rendimiento, sirviendo para establecer los planes de ahorro y, en resumen, para apoyar la defensa de nuestro dinero.

Puede ser, igualmente, el medio adecuado para introducirnos en el mundo de la Bolsa, las inversiones mobiliarias y sus cálculos financieros.

Tipos de inversión que maneja el programa

Las inversiones mobiliarias pueden subdividirse en tres grandes grupos: Títulos de Renta Fija, Títulos de Renta Variable, y Activos Monetarios.

Entre los Títulos de Renta Fija a medio y largo plazo se incluyen la Deuda Pública del Estado y Autonomías, las Obligaciones, los Bonos y las Cédulas Hipotecarias. Entre los de Renta Variable las acciones, particularmente las de sociedades con cotización en bolsa. Por último, bajo el epígrafe de Activos Monetarios o activos financieros a corto plazo, se encuentran los Pagarés del Tesoro y los Pagarés o Letras emitidos por empresas o bancos, a plazos que no suelen superar los dieciocho meses.

El equipo necesario es el siguiente: ordenador COMMODORE 64, unidad de disco 1541 y receptor de TV o monitor. Opcionalmente puede incorporarse una impresora MPS 801 o equivalente.

Una vez introducido el disco que contiene el programa en la unidad de disco, debe escribirse: **LOAD «*», 8** y, seguidamente, teclear **RUN.** Al cabo de un minuto aproximadamente, aparece la pantalla de presentación seguida del menú principal.

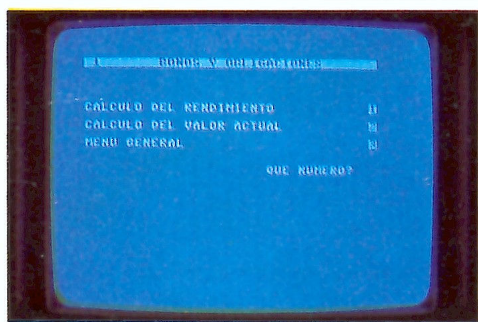
Menú general

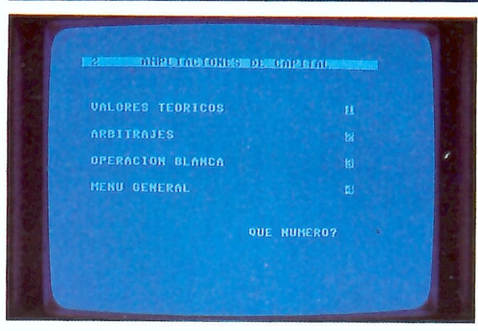
Este menú contiene los títulos de seis menús secundarios, a los que podemos acceder pulsando el número correspondiente. Además, existe una séptima opción que permite finalizar la sesión de trabajo. Así mismo, desde cada uno de los sub-menús es posible volver al menú principal pulsando el número previsto al efecto.

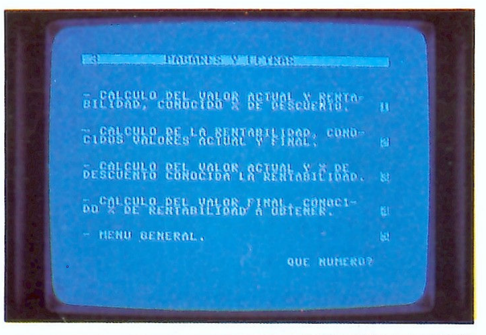
Bonos y obligaciones

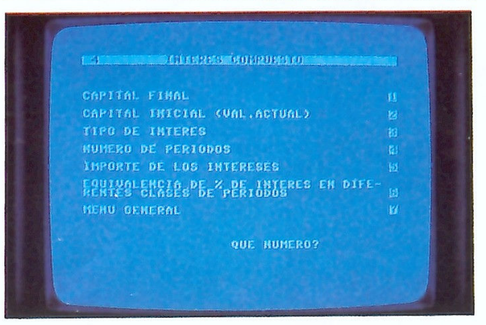
Encontramos en esta opción los cálculos del rendimiento y del valor actual para valores de renta fija a medio y largo plazo.

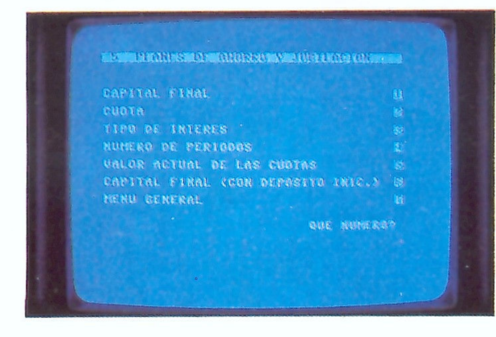












Ampliaciones de capital

Se hallan aquí los cálculos relativos a los valores teóricos de los derechos de suscripción y de las acciones cuando una sociedad amplía su capital. También se calculan los arbitrajes y la cantidad de acciones que se pueden suscribir, sin desembolso, vendiendo parte de los derechos de suscripción (operación blanca).

Pagarés y letras

Este epígrafe trata de los activos monetarios o activos financieros a corto y medio plazo. En esta opción se ofrecen todas las posibilidades de cálculo de los valores actual y final, descuento y rentabilidad efectiva de la operación.

Interés compuesto

Frecuentemente se precisa realizar cálculos de interés compuesto, que con el uso de esta opción se ven notablemente simplificados.

Planes de ahorro y jubilación

Los cálculos de anualidades, o de pagos periódicos de cuotas uniformes, tienen una amplísima gama de aplicaciones prácticas: planes sistemáticos de ahorro, fondos de jubilación, seguros de vida, amortizaciones, préstamos, etc. Esta alternativa permite diversas posibilidades en este campo.

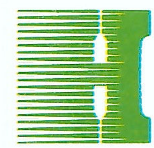
Flujo de fondos irregular

Por último, cuando los movimientos de entradas y salidas de fondos no obedecen a una regla fija, es muy difícil conocer el rendimiento real de una inversión. Disponemos en esta opción de un instrumento ideal para calcularlo, pudiendo obtener además el valor neto actual y el valor futuro.

Programa: Análisis de Inversiones
Precio: 12.000 ptas.
Programa: Referencias Bibliográficas
Precio: 9.900 ptas.
Modelo: C-64 y C-128
Soporte: Disco
Distribuidor: Casa de Software
Taquígrafo Serra, 7
Tel.: (93) 321 96 36
08029 BARCELONA

Biorritmos

¡Un gran día! Las fiestas de Navidad están próximas y... ¡atchiiis!, ¡vaya resfriado más inoportuno! Aunque no hay mal que por bien no venga: nada mejor que aprovechar la cura en casa para terminar la introducción de los 38K del excitante programa que aguarda en el cajón... ¡Un gran día!, no cabe la menor duda...! con un apagón a diez bytes de terminar el trabajo...!



Hay quien afirma que tal cúmulo de desafortunados eventos, pueden preverse con una sagaz interpretación de los ciclos biológicos del individuo. El programa que presentamos, nos resultará útil para «educar» a nuestro ordenador en estos menesteres.

El nacimiento de los biorritmos

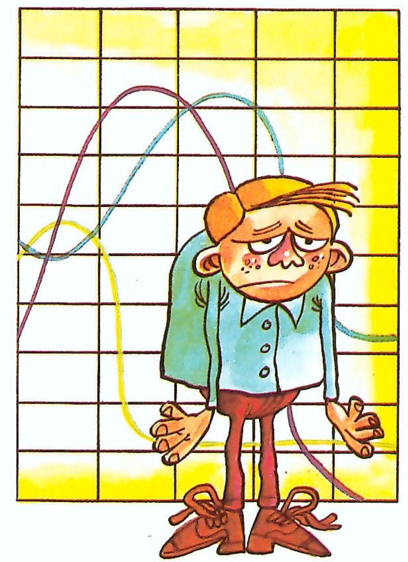
En el preciso instante de nacer, comienzan su andadura los tres ciclos biológicos que nos acompañarán durante toda la vida. Estos tres ciclos son conocidos como: físico, emocional e intelectual. Cada uno de ellos afecta a la correspondiente parcela de la actividad humana. La representación de sus trayectorias conforma sendas ondas sinusoidales que se distinguen por su diferente longitud.

La confección de un biorritmo consiste en la representación gráfica simultánea de los tres ciclos, reflejándose en el eje de abcisas (horizontal) los días del año a que pertenece cada punto representado. La interpretación del gráfico resultante difiere entre unos estudiosos y otros.

Nuestro programa traza en el centro de la pantalla una línea separadora de las zonas positivas (superior) y negativa (inferior) del biorritmo. En cualquier caso, el punto al que se presta mayor atención es aquel en el que el biorritmo pasa de la zona positiva a la negativa o viceversa (punto crítico).

¡Guárdate de los idus de marzo!

Probablemente si César hubiera hecho caso de las legendarias palabras del adivinador, la historia se habría alterado considerablemente. De forma análoga, pode-



mos utilizar nuestro programa como oráculo al que recurrir en momentos difíciles. Por supuesto que con el análisis de nuestro biorritmo no vamos a cambiar el curso de los acontecimientos, aunque, no obstante, si puede contribuir —en opinión de los expertos en el tema— a mantenernos alerta ante ciertas situaciones.

Siguiendo con las opiniones de los más versados, parece que los días más propensos al desastre son aquellos en los que más de un ciclo atraviesa un punto crítico... ¡No perdamos la calma!, la coincidencia de los tres ciclos aparece por primera vez el día en que nacemos (¡convendréis con nosotros en que fue un día bastante conflictivo!) y no se repite hasta aproximadamente 52 años más tarde.

Una última aclaración

Hay que dejar sentado que el hecho de que hoy, por ejemplo, atravesemos un punto crítico en nuestro estado físico, no quiere decir forzosamente que vayamos a contraer una enfermedad. Según los expertos, el significado a extraer de esta circunstancia es que hoy somos más vulnerables al despiadado ataque de un agente patógeno.

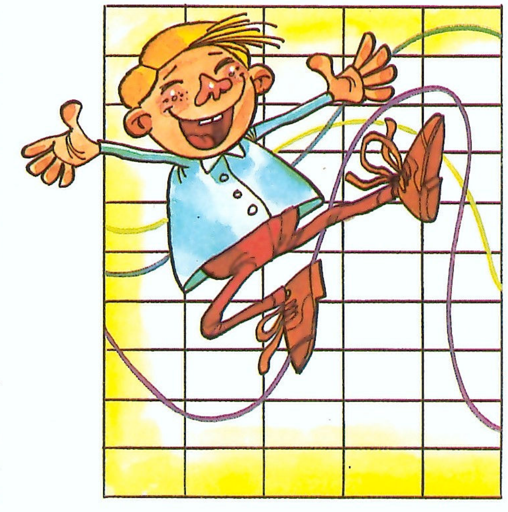
Es bien conocido el hecho de que una compañía de taxis japonesa pone en conocimiento de sus conductores el paso de los días críticos, con el fin de que estos extremen las precauciones en la conducción.

El manejo del programa

El manejo del programa es —como supondréis— extremadamente simple y, de hecho, sólo es necesario introducir la fecha de nacimiento del individuo sometido a estudio y, por supuesto, el período durante el cual deseamos conocer el biorritmo.

Las dos fechas citadas se introducen en la secuencia: día, mes, año, aportando de este último sólo las dos últimas cifras. Hay que tener en cuenta que los tres datos entran de la mano del mismo **INPUT,** de ahí que haya que separarlos por medio de las adecuadas comas. Así por ejemplo, si nuestra fecha de nacimiento es el 23 de agosto de 1961, al **INPUT** de "NACIO EL DD, MM,AA?", habrá que responder con 23, 8, 61.

Confiamos en que la suerte os acompañe durante los días críticos, y en especial en las fiestas navideñas.

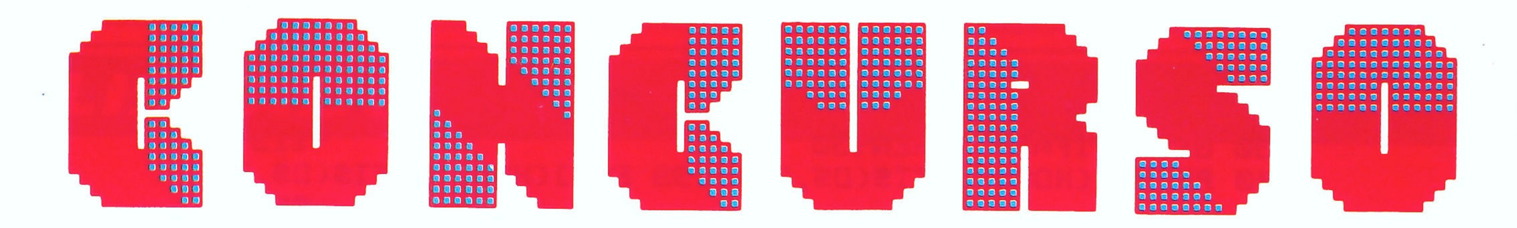


LISTADO

```
10 REM BIORRITMO # COMMODORE-64 # JUAN MANUEL LOPEZ M
ARTINEZ                                              -130-
20 POKE53280,12:POKE53281,12:H=100.01:F=.001:DIMM%(12
),M$(12)                                             -035-
30 FORI=1TO12:READM%(I),M$(I):NEXT                   -057-
40 D$="{24 ABJ}"                                     -005-
50 R$="{40 DCH}"                                     -004-
60 PI=-2*{PI }                                       -228-
70 INPUT"{CLR}{BLU}DIGAME SU NOMBRE ";N$:N$=LEFT$(N$+
"[10 ESP]",10)                                       -099-
80 INPUT"{2 ABJ}NACIO EL DD,MM,AA";DB,MB,YB          -248-
90 TB=0                                              -140-
100 T1=YB/4                                          -113-
110 IFMB<2THEN160                                    -004-
120 FORI=1TOMB-1                                     -225-
130 TB=TB+M%(I)                                      -084-
140 NEXT                                             -244-
150 IFT1<>INT(T1)ANDMD>2THENTB=TB+1                  -205-
160 TB=TB+365*YB+INT(T1)+DB                          -075-
170 INPUT"{2 ABJ}DESDE EL DD,MM,AA";D,M,Y            -093-
180 TS=0                                             -205-
190 T1=YS/4                                          -139-
200 IFM<2THEN250                                     -194-
210 FORI=1TOM-1                                      -159-
220 TS=TS+M%(I)                                      -118-
230 NEXT                                             -244-
240 IFT1<>INT(T1)ANDM>2THENTS=TS+1                   -171-
250 TS=TS+365*Y+INT(T1)+D                            -233-
260 TT=TS-TB                                         -007-
270 PS=23*(TT/23-INT(TT/23))                         -090-
280 ES=28*(TT/28-INT(TT/28))                         -095-
290 IS=33*(TT/33-INT(TT/33))                         -088-
300 GOSUB460                                         -205-
310 FORI=0TO38                                       -095-
320 P=8+8*SIN((PS+I)*PI/23)+.5                       -143-
330 PRINT"{HOM}"LEFT$(R$,I)LEFT$(D$,P+2)"{CYN}|Q|"
                                                     -186-
340 NEXT                                             -246-
350 FORI=0TO38                                       -099-
360 E=8+8*SIN((ES+I)*PI/28)+.5                       -130-
370 PRINT"{HOM}"LEFT$(R$,I)LEFT$(D$,E+2)"{YEL}|Q|"
                                                     -178-
380 NEXT                                             -250-
390 FORI=0TO38                                       -103-
400 IT=8+8*SIN((IS+I)*PI/33)+.5                      -213-
410 PRINT"{HOM}"LEFT$(R$,I)LEFT$(D$,IT+2)"{WHT}|Q|{BL
U}"                                                  -139-
420 NEXT                                             -245-
```

```
430 GETA$:IFA$=""THEN430                                   -113-
440 PRINT"{HOM}"LEFT$(D$,24)"[39 ESP]{HOM}"LEFT$(D$,2
2);                                                        -064-
450 GOTO170                                                -138-
460 PRINT"{CLR}";                                          -089-
470 PRINT"BIORRITMO:{RON}"N$"{ROF} NACIM.";                -200-
480 PRINTDB;M$(MB)" 19"MID$(STR$(YB/H+F),3,2)
                                                           -038-
490 PRINT"{BLK}<A><37 R><S>"                               -054-
500 FORI=1TO8                                              -046-
510 PRINT"<Q> |-| |-| |-| |-| |-| |-| |-| |-| |-| |-|
 |-| |-| |-| |-| |-| |-| |-| |-| <W>"                      -207-
520 NEXT                                                   -246-
530 PRINT"<Q>|C||+||C||+||C||+||C||+||C||+||C||+||C||
+||C||+||C||+||C||+||C||+||C||+||C||+||C||+||C||+||C|
|+||C||+||C||+||C|<W>"                                     -198-
540 FORI=1TO8                                              -050-
550 PRINT"<Q> |-| |-| |-| |-| |-| |-| |-| |-| |-| |-|
 |-| |-| |-| |-| |-| |-| |-| |-| <W>"                      -211-
560 NEXT                                                   -250-
570 PRINT"<Z><37 E><X>{BLU}"                               -171-
580 MD=M:D1=D:YD=Y                                         -117-
590 FORI=0TO38                                             -105-
600 IFI/2<>INT(I/2)THENPRINT"{DCH}";:GOTO670
                                                           -183-
610 A$=STR$(D1)                                            -060-
620 IFLEN(A$)<3THENB$=RIGHT$(A$,1):GOTO640                 -223-
630 B$=MID$(A$,2,1)+"{ABJ}{IZQ}"+RIGHT$(A$,1)+"{ARB}"
                                                           -200-
640 PRINTB$;                                               -232-
650 C$="{ROF}{7 IZQ}{ARB}"                                 -017-
660 IFD1=1ORD1=2THENPRINT"{IZQ}{ABJ}{RON}"M$(MD)" 19"
MID$(STR$(YD/H+F),3,2)C$;                                  -053-
670 D1=D1+1                                                -064-
680 IFD1<=M%(MD)THEN730                                    -088-
690 IFMD<>2THEN720                                         -083-
700 IFYD/4<>INT(YD/D)THEN720                               -212-
710 IFD1<30THEN730                                         -036-
720 D1=1:MD=MD+1:IFMD>12THENMD=1:YD=YD+1                   -199-
730 NEXT                                                   -249-
740 PRINT"{3 ABJ}":PRINT" FISICO={CYN}|Q|{BLU} {DCH}
EMOCIONAL={YEL}|Q|{BLU}[3 ESP]INTELECTUAL={WHT}|Q|{BL
U}";                                                       -057-
750 C$="ALTO[2 ESP]CRITICO[2 ESP]BAJO                      -091-
760 PRINT"{HOM}{2 ABJ}{IZQ}";                              -155-
770 FORI=1TO20                                             -097-
780 PRINTMID$(C$,I,1)"{ABJ}{IZQ}";                         -001-
790 NEXT                                                   -255-
800 RETURN                                                 -152-
810 DATA31,ENE,28,FEB,31,MAR,30,ABR,31,MAY,30,JUN
                                                           -065-
820 DATA31,JUL,31,AGO,30,SEP,31,OCT,30,NOV,31,DIC
                                                           -097-
```

CONCURSO





El travieso C-Byte tiene el honor de invitaros a la participación en nuestro quinto concurso de programación. Los requisitos necesarios son bien pocos:

- Saber programar un ordenador COMMODORE.
- Ser español o extranjero y
- Tener una edad comprendida entre 5 y 105 años.

Fácil, ¿verdad?
En cuanto a los premios, la mar de atractivos:

dotado con un gran premio de

100.000 Ptas.

en material informático, a escoger por el premiado.

Y en fin, si alguno de los programas destaca por su originalidad, estética o comicidad, no sería de extrañar que le cayera alguna cosilla más...

BASES DEL CONCURSO

1. Los programas remitidos al concurso deberán ser creación original del autor o autores, y completamente inéditos, pudiendo remitir tantos programas como se desee.

2. Los programas deberán ser enviados en casete o disco flexible a TU MICRO COMMODORE (Concurso de programación). Apartado de Correos 61.294. 28080 MADRID.

3. Los programas podrán ser de cualquier tipo (juegos, utilidades, gestión, educativos) y habrán de estar escritos en lenguaje BASIC o Código Máquina.

4. Los programas deberán ser remitidos desprovistos de cualquier tipo de protección, que impida o dificulte el análisis del mismo, así como reproducción en estas páginas y su introducción como listado siguiendo el sistema FUERA ERRORES.

5. Cuando la ejecución del programa precise de la concurrencia de determinado periférico o aditamento (joysticks, tabletas gráficas, programas comerciales de ayuda), se valorará la indicación de las modificaciones pertinentes, para que el programa pueda ser disfrutado por cualquier usuario en la configuración básica.

6. Todo programa presentado al concurso deberá acompañarse de los siguientes datos:

- Datos personales del concursante.
- Nombre del programa.
- Modelo para el que está destinado.
- Breve descripción del programa detallando las indicaciones necesarias para su ejecución.

7. Los programas premiados pasarán a ser propiedad de la revista TU MICRO COMMODORE, pudiendo hacer ésta libre uso de ellos, y renunciando sus autores a cualquier otra compensación distinta al premio.

8. Los programas no premiados que por su calidad se hagan merecedores de su publicación, serán adquiridos por la editorial, aplicando la tarifa vigente.

9. Los programas recibidos con posterioridad a la fecha tope de la presente edición, serán automáticamente incluidos en los destinados a la siguiente.

10. El jurado decidirá sobre todos los aspectos no contemplados en estas bases y su decisión será inapelable.

Y ahora a darse mucha prisa, el plazo para la recepción de programas termina el próximo día 1 de Marzo de 1987.

¡¡SUERTE!!

CURSO DE BASIC

UTILIZANDO EL COLOR

El color es uno de los atributos que hacen del COMMODORE 64 un ordenador atractivo y versátil, manteniendo una indudable sencillez de manejo. Hasta ahora no sabemos hacer uso de esta importante herramienta, pero en el transcurso de los siguientes capítulos, perfilaremos los conocimientos necesarios como para empezar a explotar las posibilidades cromáticas de nuestro ordenador, sacándole partido a esta nueva e importante faceta del mismo.

Lo primero que debemos saber sobre este tema es que, a efectos de color, la pantalla de nuestro monitor o receptor de TV, se encuentra dividida en dos zonas bien diferenciadas, cuya denominación podría ser: «primer» y «segundo» plano; o zona circundante y zona central de la pantalla.

La zona de pantalla sobre la cual podemos imprimir es una porción rectangular de 25 filas de 40 columnas (1000 posiciones). El resto de la misma, es decir, la parte que se encuentra por encima o por debajo, o la izquierda o derecha de los márgenes citados, no es accesible a través de la sentencia **PRINT.**

Esta zona no utilizable para la escritura, conforma una especie de «fondo» o segundo plano, bordeando la pantalla, y puede tomar cualquiera de los 16 colores de que dispone el ordenador.

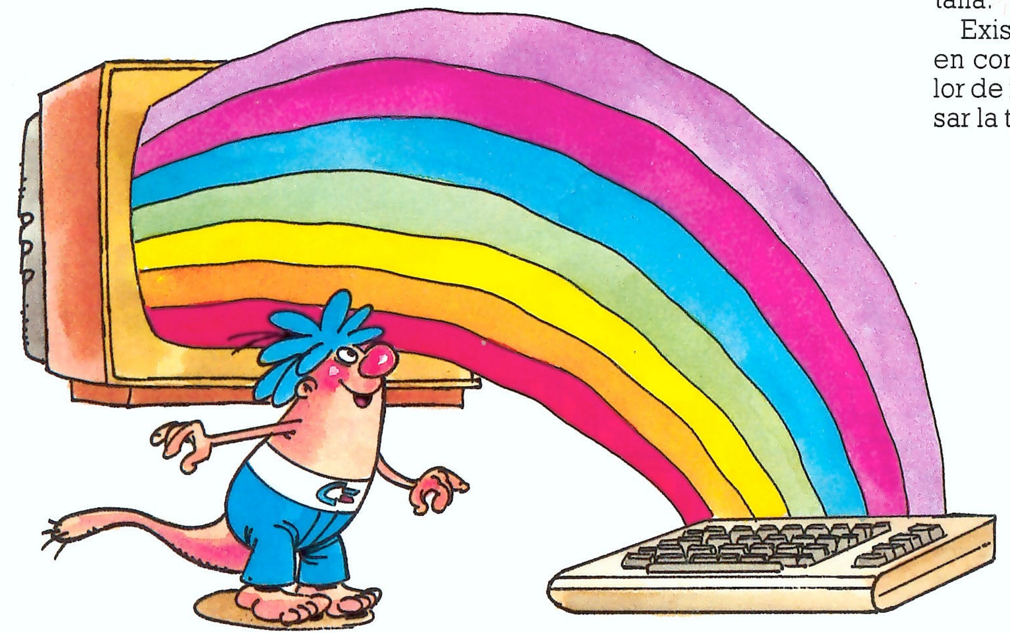
PRIMER PLANO DE IMPRESION

Sobre los colores del primer, segundo y tercer plano aparecidos en la pantalla a la hora de realizar la impresión, podemos efectuar algunas comparaciones.

Por ejemplo, el color del primer plano de impresión representa el de la «tinta», con la cual escribimos **(PRINT),** correspondiendo entonces el color del segundo plano al del papel sobre el que realizamos la impresión y, por último, el tercer plano al del color de la mesa sobre la que nos apoyamos, aunque no podemos utilizar este último espacio para la impresión de caracteres.

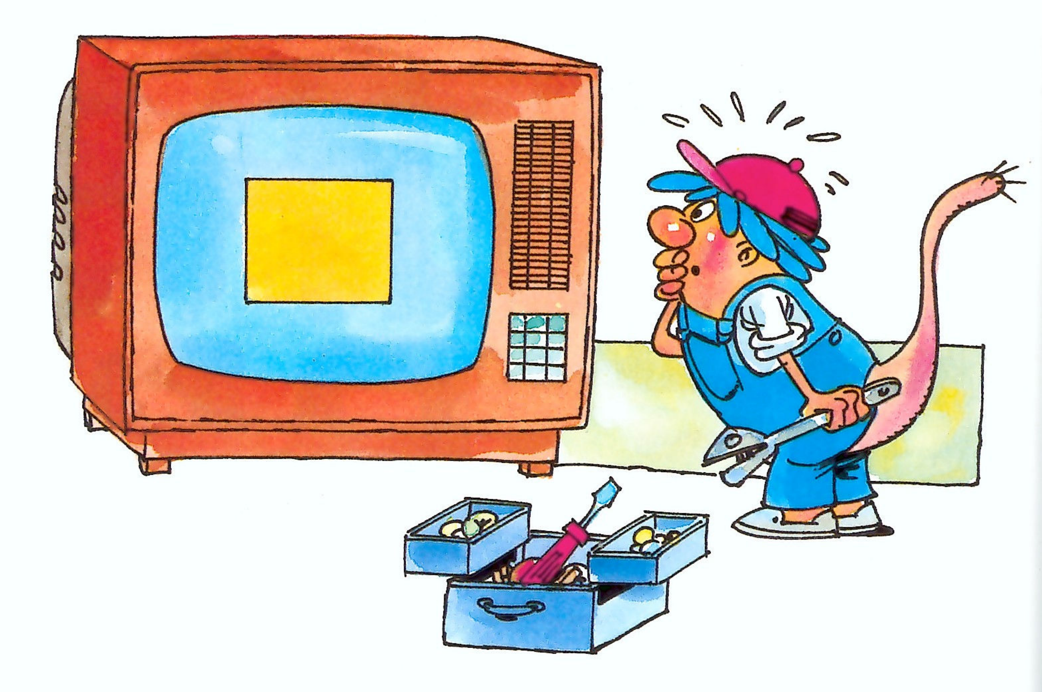
Por otro lado, y aunque hasta el momento no nos hemos decidido a hablar del color en nuestro ordenador y de la forma de tratarlo, es evidente que, desde el mismo momento del encendido, éste toma ciertos valores por «defecto» para los colores de los tres planos; correspondiéndose precisamente éstos, por omisión, con los que obtenemos al pulsar la combinación de teclas **RUN STOP** y **RESTORE,** uno de cuyos cometidos es, precisamente, retornar a los valores iniciales los parámetros de color de la pantalla.

Existe un procedimiento inmediato, en comando directo, de alterar el color de impresión. Este consiste en pulsar la tecla **CTRL** en combinación con un número del 1 al 8. Con ello, conseguimos que el cursor continúe parpa-



El color es uno de los atractivos que hacen del Commodore 64 un ordenador atractivo y versátil.

La pantalla de nuetro T.V., se encuentra dividida en dos zonas: circundante y central de la pantalla.



deando en el nuevo color seleccionado y, además, que todos los caracteres tecleados a continuación adopten un color característico de la combinación de teclas pulsadas.

Sin embargo, de esta forma sólo podemos acceder a un juego de 8 colores. Por ello, existe una segunda combinación de teclas que podemos seleccionar. Esta combinación es la formada por la tecla **COMMODORE** y un número del 1 al 8. De esta forma, si tenemos en cuenta las combinaciones posibles a través de las teclas **CTRL** y **COMMODORE,** encontraremos fácil acceso a la gama completa de 16 colores que soporta nuestro ordenador.

Dicho esto, ha llegado el momento de que efectuemos algunas prácticas de cambio del color de impresión en pantalla. Quede claro que si, en algún momento, vemos desaparecer el cursor, esto se debe a que hemos adoptado el mismo color para el primer plano que para el existente por defecto en el fondo inmediato. Para solucionar este problema, basta con seleccionar un nuevo color u optar por pulsar la combinación de teclas **RUN STOP** y **RESTORE,** las cuales borran la pantalla y retornan los valores de color a los iniciales.

CAMBIO DE COLOR EN PROGRAMA

Es evidente que, de no poderse utilizar el color más que en el modo directo, no nos resultaría de gran utilidad, por el mismo motivo que es mucho más interesante la confección de programas BASIC, que el empleo del comando directo.

Para introducir los cambios de color dentro de nuestros programas, basta con recurrir a un sistema parecido al ya descrito para la inclusión en forma de cadena de caracteres de los desplazamientos del cursor, el posicionamiento en el origen de la pantalla, o la situación con borrado previo.

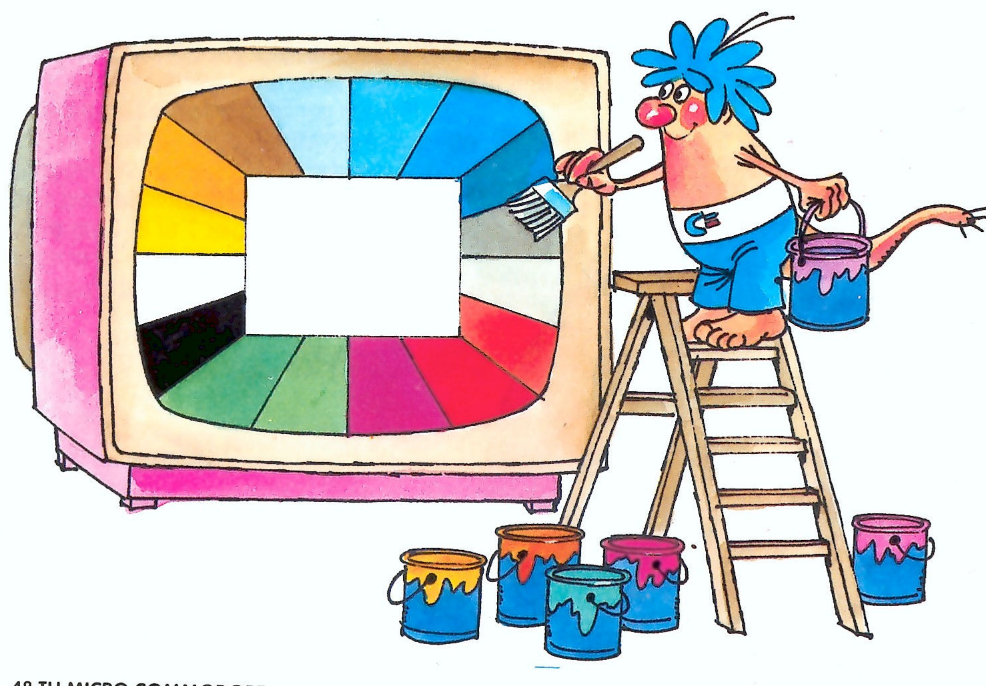
Basta, pues, con incluir el carácter generado por la pulsación de **CTRL** o **COMMODORE** y algún número de 1 a 8, entre comillas, justo antes del punto a partir del cual deseamos que la impresión por medio de **PRINT** se produzca en un color diferente al anterior.

De esta forma, el nuevo color seleccionado continuará vigente, hasta que se halle en el programa otro código de cambio de color que modifique esta situación. Es más, una vez concluido el programa, el último color de «tinta» seleccionado permanecerá activo, hasta que pulsemos la combinación de teclas necesaria para reinicializar los valores por defecto.

En el siguiente cuadro se describen las correspondencias entre las teclas de control y los colores que representan:

TECLAS	COLOR	TECLAS	COLOR
CTRL 1	NEGRO	COMMODORE 1	NARANJA
CTRL 2	BLANCO	COMMODORE 2	MARRON
CTRL 3	ROJO	COMMODORE 3	ROJO CLARO
CTRL 4	AZUL CLARO	COMMODORE 4	GRIS 1
CTRL 5	PURPURA	COMMODORE 5	GRIS 2
CTRL 6	VERDE	COMMODORE 6	VERDE CLARO
CTRL 7	AZUL	COMMODORE 7	AZUL CLARO
CTRL 8	AMARILLO	COMMODORE 8	GRIS 3

Los 16 colores de que se compone la «paleta» de nuestro ordenador, los identificaremos de ahora en adelante por los números 0 al 15; correspondiéndose los ocho primeros al número del que acompañamos la tecla **CTRL** menos uno. Asimismo, los colo-



La zona no utilizable para la escritura puede tomar cualquiera de los 16 colores de que dispone el ordenador.

*La combinación de las teclas **RUN STOP** y **RESTORE,** tiene el cometido de retornar a los valores iniciales los parámetros de color de la pantalla.*



res 8 a 15 son los obtenidos por pulsación de la tecla **COMMODORE,** y el número correspondiente de la fila superior del teclado más siete.

IMPRESION EN VIDEO INVERSO

Cuando nos encontramos en el modo normal de impresión en pantalla, los caracteres aparecen en el color seleccionado mediante el carácter de control correspondiente, o en el asignado automáticamente al encendido del ordenador, pero siempre sobre el color de fondo de la pantalla.

Como sabemos, los caracteres que imprimimos en pantalla están formados por una matriz de puntos de 8x8, es decir, una superficie de 64 puntos en total. Por supuesto, ningún carácter emplea la totalidad de ellos; de esta forma, los empleados en la composición del carácter adoptan el color de la «tinta», mientras que el resto de los componentes de la matriz toman el color del «papel» o segundo plano.

Pues bien, existe un método de invertir este estado de cosas denominado impresión en «vídeo inverso». Dentro de este modo, los puntos de la matriz empleados en la definición del carácter adoptan el color del fondo, mientras que el resto toman el del carácter (tinta). Con ello, se produce un efecto de resalte en pantalla, adecuado para cabeceras o determinados mensajes.

Para acceder a esta forma de impresión, dentro del modo comando, basta con pulsar **CTRL** y 9. De forma análoga, y para regresar al modo de impresión normal, teclearemos **CTRL** y 0. Podemos probar ahora a escribir algunas cosas en la pantalla, haciendo uso de la conexión y desconexión de la impresión en vídeo inverso.

Por supuesto, la posibilidad de empleo del vídeo inverso no está restringida al modo directo. Podemos emplearla también dentro de programas, incluyendo los caracteres de conexión y desconexión dentro de comillas; de forma similar a lo comentado para los caracteres de control del color.

Como caso particular debemos advertir que, cuando pulsamos la barra espaciadora dentro del modo de vídeo inverso, obtenemos una barra del color de la «tinta» especificada para la impresión de caracterese. Apoyándonos en esta propiedad, podemos codificar un pequeño programa que nos sirva de «carta de ajuste» de color, al comienzo de la sesión:

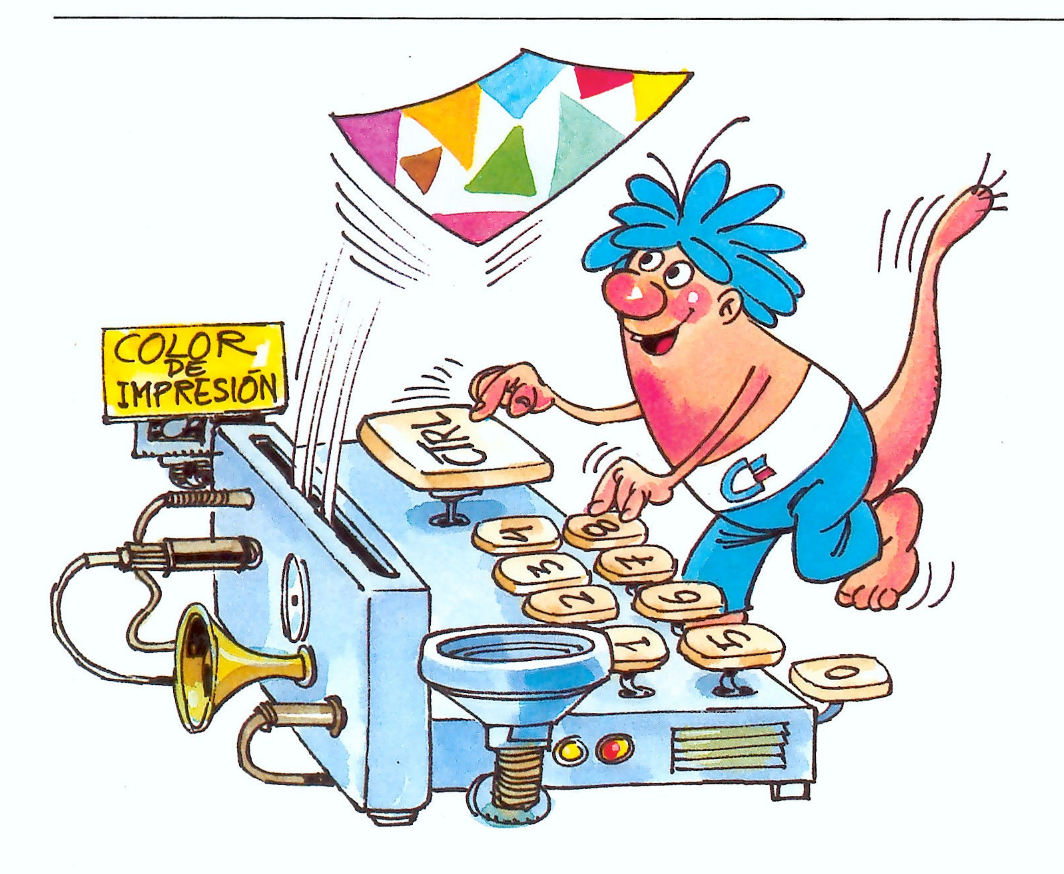
LISTADO 1

Para realizar este programa, almacenamos en la variable X$ una cadena de caracteres compuesta por un código de comienzo de vídeo inverso, cinco espacios en blanco y el correspondiente código de fin de vídeo inverso.

A continuación, escribimos una línea en la que colocamos los primeros ocho códigos de color seguidos de la variable X$, varias veces hasta completar la primera mitad de la pantalla. Seguidamente, repetimos la operación para los restantes ocho códigos de color, obteniendo como resultado la representación en pantalla,



A la hora de realizar la impresión podemos representar el color del primer plano como la tinta, del segundo como el papel, y del tercer plano como la mesa sobre la que nos apoyamos.



*Para alterar el color de impresión en comando directo, pulsaremos la tecla **CTRL** en combinación con un número del 1 al 8.*

como masas de color, de los 16 posibles en nuestro ordenador.

COMBINACIONES DE COLOR

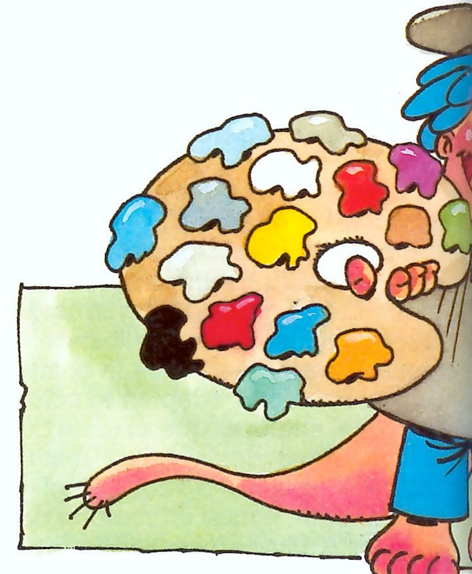
En teoría, cualquier combinación de colores de carácter y fondo, siempre que éstos no sean iguales, debería ser válida para la representación en pantalla.

Sin embargo, es obvio que algunas de las combinaciones posibles no son de lo más acertadas, produciendo un cierto efecto de distorsión de la imagen, que dificulta la lectura. Este hecho está motivado por la definición de la pantalla de TV con que trabajamos, la cual encuentra dificultad al representar puntos próximos de colores ciertamente incompatibles.

Por otro lado, es evidente que el hecho de elegir colores muy próximos en tonalidad para el primer y segundo plano de impresión, no puede más que acarrearnos problemas de poca legibilidad de los datos impresos.

Además, está comprobado que algunas combinaciones de color resultan más agradables para la vista, sobre todo cuando vamos a trabajar varias horas delante de la pantalla del receptor de TV.

Por ello, el ordenador selecciona de inicio unos colores para los tres planos que facilitan la lectura, tanto para receptores de TV en color como en blanco y negro. En resumen, podemos decir que existen combinaciones de color excelentes, buenas y regulares.



Sólo resta decir, en cuanto a los códigos de color de impresión, que pueden ser invocados igualmente a través de su posición dentro del código ASCII, mediante la función BASIC **CHR$** o bien con POKE 646, color (0-15). De esta forma, disponemos de una nueva posibilidad a la hora de codificar programas que contienen especificaciones de color.

SEGUNDO Y TERCER PLANO DE IMPRESION

Ya conocemos la forma de dotar la impresión del color que deseemos, a través de los códigos de control. Debemos saber, además, como alterar

Si en algún momento vemos desaparecer el cursor, esto se debe a que los colores de fondo y primer plano son el mismo.



Los caracteres que imprimimos en la pantalla están formados por una matriz de puntos 8x8.

A través de las combinaciones posibles de las teclas ***CTRL*** *y* ***COMMODORE,*** *tendremos acceso a la gama de 16 colores que soporta nuestro ordenador.*



los colores del segundo y tercer plano de impresión.

Dos son las posiciones de memoria concretas que controlan los colores de fondo del segundo y tercer plano, repectivamente las direcciones decimales 53281 y 53280. En ellas, se encuentra almacenado permanentemente el código de color correspondiente (0—15).

Por ello, basta efectuar un **POKE** a la dirección adecuada, con el valor del código deseado, para que el color del plano correspondiente se vea afectado.

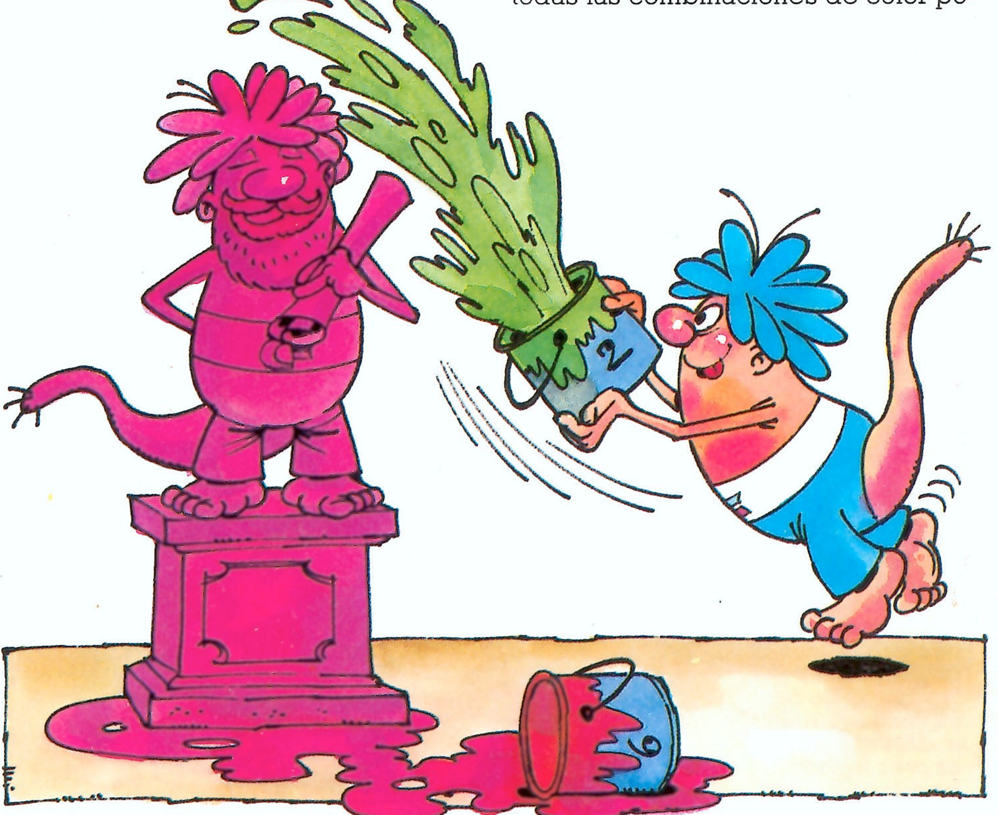
Además, y de forma análoga a como sucede con el primer plano de impresión, los valores asignados a estas posiciones permanecen en lo sucesivo, aunque el programa llegue a su fin.

Debido a esta circunstancia, debemos tener en cuenta la posibilidad de restaurar estos valores a los iniciales antes de concluir el programa, u optar por pulsar **RUN STOP** y **RESTORE** para hacerlo.

En el siguiente cuadro se relacionan los diferentes argumentos de la sentencia **POKE** y los colores generados para cualquiera de los dos planos:

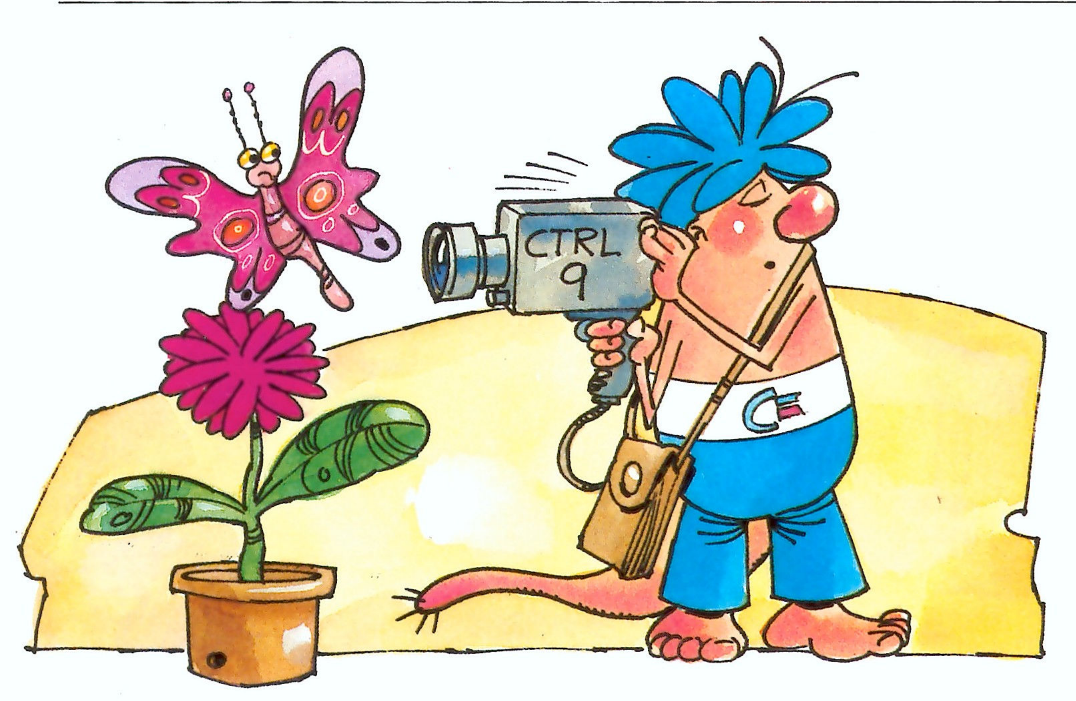
CODIGO	COLOR
0	NEGRO
1	BLANCO
2	ROJO
3	AZUL CLARO
4	PURPURA
5	VERDE
6	AZUL
7	AMARILLO
8	NARANJA
9	MARRON
10	ROJO CLARO
11	GRIS 1
12	GRIS 2
13	VERDE CLARO
14	AZUL CLARO
15	GRIS 3

Para practicar un poco con todo esto, proponemos la codificación de un pequeño programa que nos muestra todas las combinaciones de color po-



Una vez que seleccionemos un color, este permanecerá vigente hasta introducir un código de cambio de color que modifique esta situación.

*Para acceder a la impresión en vídeo inverso, debemos situarnos en modo comando y pulsar **CTRL** y 9.*



sibles de los planos segundo y tercero:

LISTADO 2

La línea 50 nos presenta un curioso sistema de averiguar el contenido de la primera mitad de un byte. Podemos considerar que un BYTE (8 bits), está formado a su vez por dos NIBBLEs (4 bits). Pues bien, como los códigos de color no superan el valor de 15, pueden almacenarse codificados en binario, con una ocupación de un único nibble.

De esta forma, la expresión **PEEK () AND 15** nos da como resultado directo un número de 0 a 15, precisamente por coincidir la representación en binario del 15 con 00001111. En este caso, la operación **AND** realiza un **AND** «lógico» bit a bit entre el contenido de la posición de memoria a comprobar y la máscara 00001111 descrita.

La línea 60 de programa tiene como único cometido el servir de retardo a la aparición de nuevos colores. Podemos, por lo tanto, retocarla a nuestro antojo.

LA MEMORIA DE PANTALLA

Ya hemos dicho con anterioridad que podemos referirnos a la pantalla como una matriz de 25 filas por 40 columnas. Pues bien, esto es más cercano a la realidad de lo que parece.

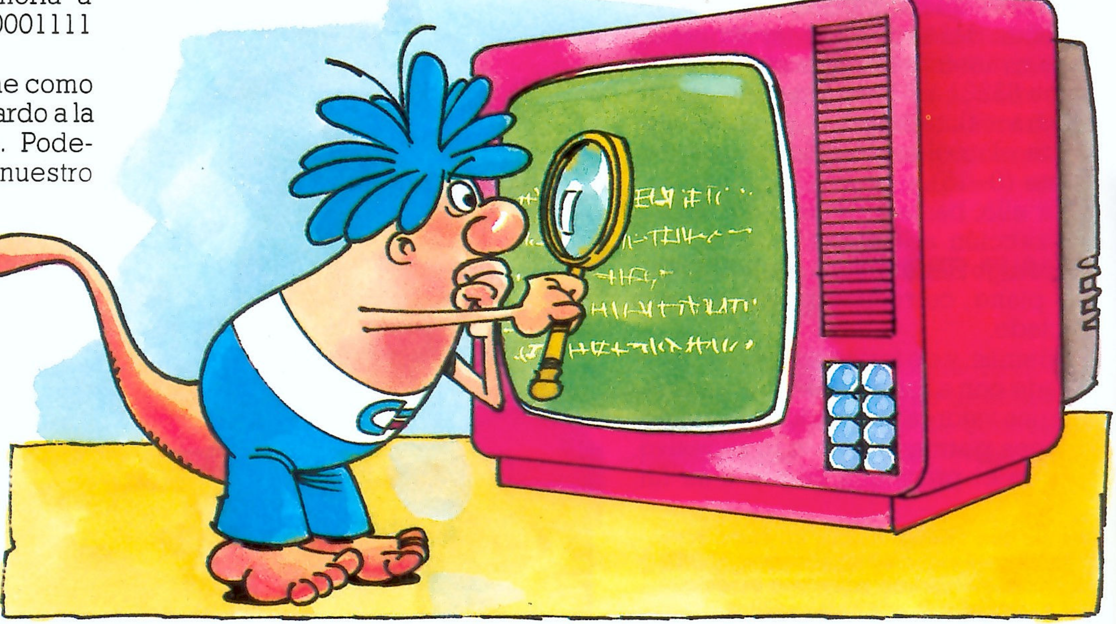
Lo que nosotros vemos en la pantalla en un momento determinado, como resultado de las diferentes impresiones por medio de la sentencia **PRINT,** es almacenado en una zona especial de la memoria RAM denominada «memoria de pantalla». Esta memoria está compuesta por 1000 posiciones consecutivas, concretamente las que van desde la dirección decimal 1024 a la 2023.

Por supuesto, cada una de estas posiciones es capaz de albergar un byte, o lo que es lo mismo, uno cualquiera de los caracteres que podemos encontrar en el teclado, o alguno de los únicamente accesibles mediante la función **CHR$.**

Cuando, por ejemplo, borramos el contenido de la pantalla por medio del carácter de control BASIC, lo que realmente estamos haciendo es colocar un espacio en blanco en cada una de las 1000 posiciones de la memoria de pantalla. Generamos, por lo tanto, mil instrucciones simples a ejecutar por el intérprete BASIC.

En general, cada vez que recurrimos a la sentencia **PRINT,** el intérprete no hace más que trabajar directamente en la memoria de pantalla a partir de los datos que le suministramos como parámetro de la sentencia.

A pesar de que la sentencia **PRINT** puede ahorrarnos mucho trabajo de codificación, no cabe duda que podemos trabajar directamente en la propia memoria de pantalla a través de la sentencia **POKE.** Sin embargo, debemos recurrir a un pequeño algoritmo que nos permita mantener la disposición de «filas» y «columnas» a la que estamos acostumbrados.



Si elegimos colores muy próximos en tonalidad para el primer y segundo plano de impresión, puede acarrearnos problemas de poca legibilidad de los datos impresos.

*Basta con efectuar un **POKE** a la dirección adecuada, y con el código deseado para alterar el color del segundo y tercer plano.*

La fórmula general para acceder a una posición concreta de la pantalla, podemos expresarla como:

P=1024+F*40+C

Donde F representa el número de fila (0-24) y C el de columna (0-39).

Sin embargo, dentro de esta estructura nos hemos olvidado del color, entre otras cosas, porque en un sólo byte no es posible almacenar más información que la relativa a la formación del carácter.

Para resolver este inconveniente, existe una memoria «gemela» de la de pantalla denominada de «color». Esta zona de la RAM comienza en la posición decimal 55296, y consta de 1000 posiciones; donde cada una de ellas mantiene su homóloga dentro de la memoria de pantalla.

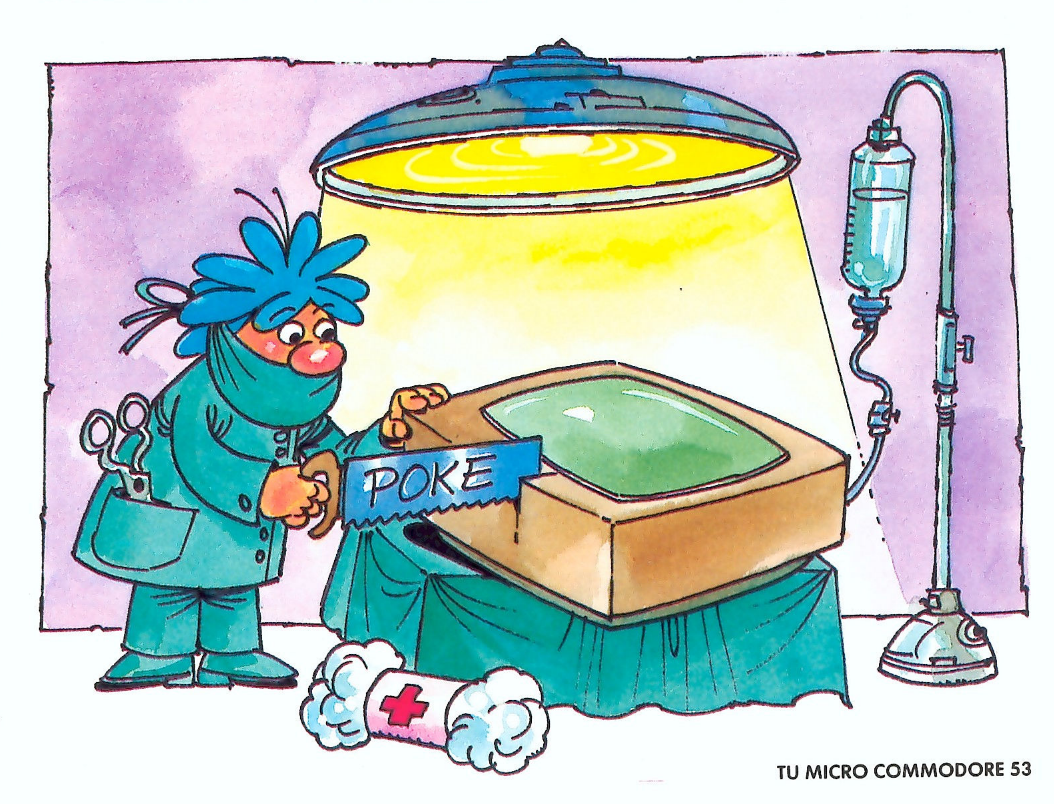
La fórmula general para acceder a una posición concreta de la memoria de color, podemos expresarla como:

P=55296+F*40+C

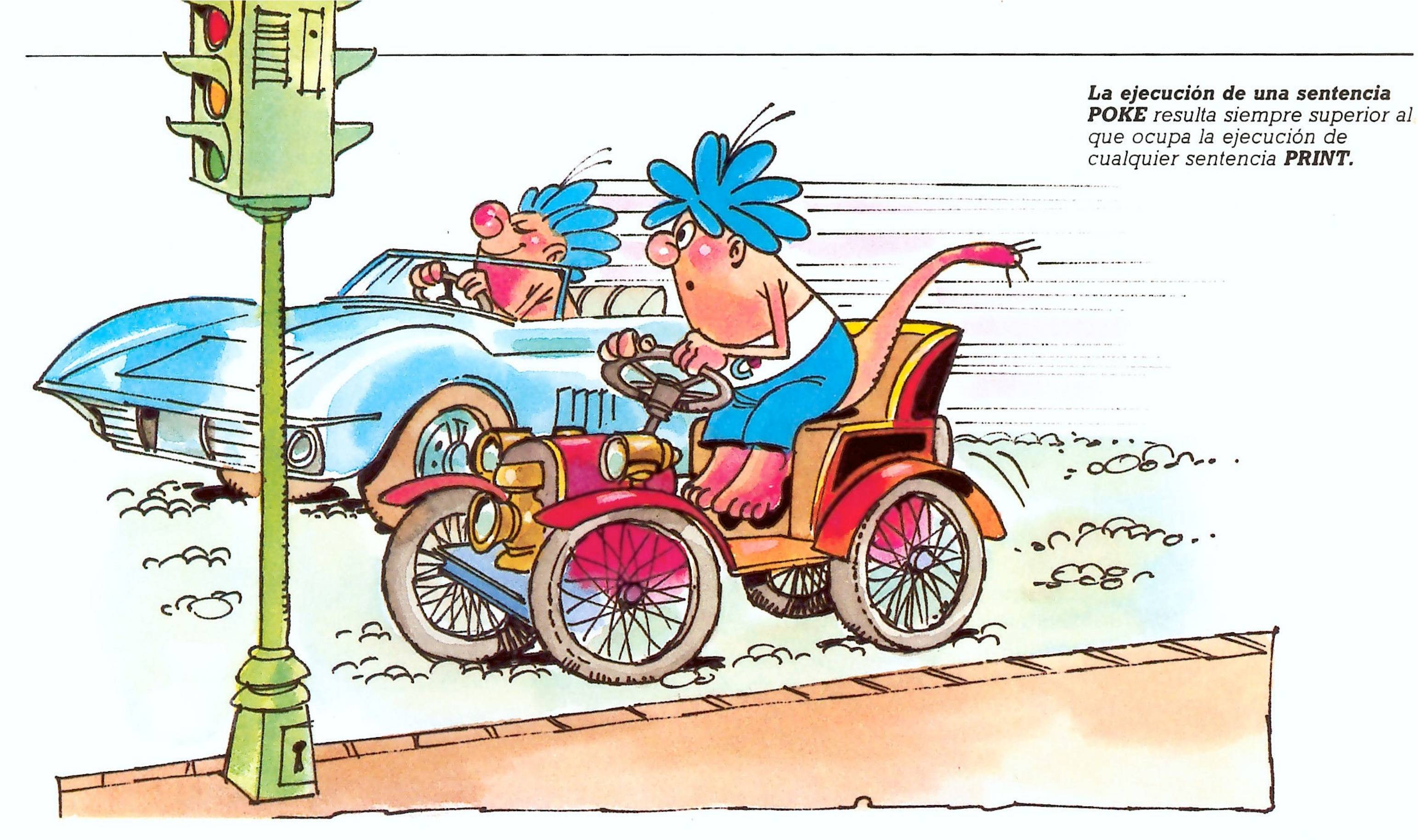
Donde F representa el número de fila (0-24) y C el de columna (0-39). En general, deberemos efectuar los **POKE** por parejas, uno hacia cada correspondiente zona de memoria, cuando deseemos hacer aparecer determinado carácter con un color concreto, en una posición de pantalla.

El tratamiento de la pantalla por medio de la sentencia **POKE** ofrece considerables diferencias frente al uso convencional **(PRINT).** En general podemos decir que la primera de ellas es que los **POKEs** a las memorias de pantalla no influyen sobre el posicionado del cursor.

Esto es lo mismo que decir que la única forma de escribir un carácter en el ángulo inferior derecho de la pantalla es, precisamente, mediante una sentencia **POKE** dirigida a la memoria de pantalla; ya que si intentásemos



*El tratamiento de la pantalla, también, puede realizarse mediante **POKE.***



La ejecución de una sentencia POKE** resulta siempre superior al que ocupa la ejecución de cualquier sentencia **PRINT.

LISTADO 1

```
10 REM - CARTA DE AJUSTE                              -210-
20 X$="{RON}[5 ESP]{ROF}":PRINT"{CLR}";               -156-
30 FOR I=1 TO 11                                      -134-
40 PRINT"{BLK}"X$"{WHT}"X$"{RED}"X$"{CYN}"X$"{PUR}"X$
"{GRN}"X$"{BLU}"X$"{YEL}"X$;                          -019-
50 NEXT                                               -196-
60 FOR I=1 TO 11                                      -137-
70 PRINT"{NRJ}"X$"{MRN}"X$"{RCL}"X$"{GR1}"X$"{GR2}"X$
"{VCL}"X$"{ACL}"X$"{GR3}"X$;                          -248-
80 NEXT                                               -199-
```

LISTADO 2

```
10 REM - SEGUNDO Y TERCER PLANO                       -223-
20 FOR I=0 TO 15                                      -136-
30 FOR J=0 TO 15                                      -138-
40 POKE 53280,I:POKE 53281,J                          -076-
50 PRINT "{CLR}MARCO=" PEEK(53280) AND 15 "FONDO=" PE
EK(53281) AND 15                                      -242-
60 FOR K=0 TO 499:NEXT                                -071-
70 NEXT J,I                                           -165-
```

LISTADO 3

```
10 REM - RELLENO DE PANTALLA                          -249-
20 PRINT"{CLR}";                                      -033-
30 FOR I=1 TO 999                                     -207-
40 PRINT "-";                                         -221-
50 NEXT                                               -196-
60 PRINT"{HOM}";                                      -165-
70 FOR I=1024 TO 2023                                 -133-
80 POKE I,43                                          -179-
90 NEXT                                               -200-
```

hacerlo mediante un **PRINT** convencional, la pantalla efectuaría un «scroll» hacia arriba, automáticamente.

Por otro lado, el tiempo que le requiere al BASIC la ejecución de una sentencia **POKE,** resulta siempre superior al que ocupa la ejecución de la sentencia **PRINT.** Esto puede que no tenga mayor importancia en condiciones normales, pero si llega a ser elemento de decisión, cuando nos encontramos en la necesidad de obtener la mayor velocidad de ejecución posible, por ejemplo, en la codificación de un juego de acción que se desarrolla en la pantalla.

Para mayor evidencia de lo dicho, proponemos efectuar una misma tarea repetitiva, haciendo uso de las dos posibilidades de que disponemos para dirigirnos a la pantalla. Esta tarea puede ser, por ejemplo, llenar la pantalla, carácter a carácter para comparar velocidades de ejecución, hasta completar sus 1000 posiciones:

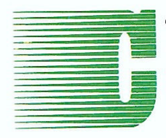
LISTADO 3

Como hemos podido apreciar en este breve programa, se nota la mayor velocidad de la sentencia **PRINT.** Sin embargo, más adelante tendremos oportunidad de ver que la elección de uno de los dos métodos de impresión, o incluso la combinación de los dos, son fruto de un análisis muy concreto de la función que pretendemos realizar.

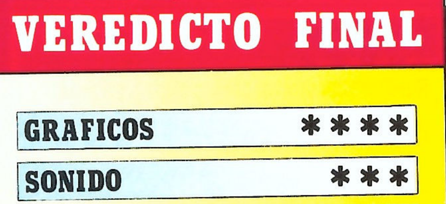
FIONA
Una bruja alocada

Fiona

Cuando éramos unos tiernos infantes e íbamos por la vida descubriendo cosas nuevas y excitantes, siempre había algunas que se nos prohibían, pero nosotros con la cabezonería típica de un pequeño, volvíamos a esa cosa vetada una y otra vez, hasta que alguien nos decía: «si tocas eso o haces aquello va a venir "tal cual" y te va a llevar»; ¿Quién no recuerda aquellos personajes terribles de nuestra infancia tales como el hombre del saco (con sus variantes del hombre de la bolsa, oscuro, negro, sucio...), la bestia parda y algunos más?; pero sin duda el más terrorífico era la bruja (solía llamarse Piruja o negra); así que para rechazar todos esos miedos aquí tenemos, ante nuestros ojos, y en nuestras manos a... Fiona, la bruja alocada.


Como toda bruja que se precie, debe presentar los atributos de pelo negro y enmarañado, largo por supuesto; gran sombrero de pico negro, como su vestido, mirada honda y escrutadora, y lo más importante e imprescindible: la escoba; sin ella una bruja no sería tal. Pues Fiona responde a este cuestionario de la «perfecta bruja asusta-niños» en todas sus características, pero además ocurre que es... un poco loca.

Las demás brujas de la A.B.U.E.L.A. (Asociación de Brujas Unicas, Esperpénticas, Latosas y Antidiluvianas), no están nada contentas con Fiona, ya que ésta se ha lanzado a hacer travesuras sin fin, y su gran actividad no responde a la apatía general de que hacen gala sus compañeras (deben cumplir lo de Antidiluvianas, y ello conlleva estar ya muy cansada); por esta causa, han decidido echarla de la asociación sea como sea.



VEREDICTO FINAL	
GRAFICOS	* * * *
SONIDO	* * *
ORIGINALIDAD	* * * * *
DIFICULTAD	* * * *
INTERES	* * * *

La sorpresa de Fiona al remontarse por los cielos con su escoba como cada día es mayúscula, ya que sus compañeras la atacan a lomos de sus respectivas, lanzándola terribles hechizos para mandarla al infierno, de donde es muy difícil salir, incluso para cualquier bruja. Así que nuestra amiga se va a tener que defender de estos ataques poniendo en juego toda su agilidad y rapidez, lanzando sus propios conjuros contra sus (isniff!) antiguas amigas, pues la cosa va realmente en serio, y Fiona no quiere ser la primera en caer a los abismos de fuego, ¿quieres tú ayudarla?







FICHA TECNICA	
NOMBRE	FIONA: UNA BRUJA ALOCADA
PRECIO	2.300 PTAS.
SOPORTE	CINTA
TIPO	ACCION
MODELO	C-64, C-128
OBSERVACIONES	¿Y LAS INSTRUCCIONES?



Thrust

Después de una prolongada paz galáctica, el año 3000 ha marcado un peligro inequívoco para los planetas pacíficos de la Confederación Estelar, miles de naves procedentes de las galaxias próximas y representando a sus respectivos mundos se han unido en el Imperio Intergaláctico, buscando conquistar al resto de planetas libres. Su ferocidad y ansia de poder se ha hecho sentir rápidamente, habiendo caído ya bajo su yugo algunas estrellas de la periferia de nuestra Galaxia Neutron, además de otras muchas que resisten a duras penas, la única solución es... Thrust.



Se ha formado un cuerpo de la resistencia, que tiene como misión sabotear los planes enemigos, que habiendo estudiado los objetivos a seguir, ha elegido como prioritario el robo de los núcleos Klystron de los planetas de abastecimiento del Imperio Intergaláctico. Estos núcleos se encuentran en determinados planetas donde el enemigo tiene reservas de fuel y armamento, si se consigue sacarlos de la gravedad de ese astro, la estación de abastecimiento será destruída.

La nave elegida para el robo está dotada de lasers, así como de potentes motores triptógenos de antigravedad, lo que permitirá elevar el núcleo hasta el espacio exterior. Los planetas en cuestión están defendidos por baterías térmicas que dispararán en cuanto detecten la nave. La energía de estas baterías está mantenida por un reactor nuclear; impactando en él, es posible poner fuera de combate a las baterías durante un corto espacio de tiempo, aunque excesivos impactos harán que entre en fase crítica y se destruya en 10 segundos con todo el planeta, dando al traste con nuestra misión, al no poder rescatar el núcleo Klystron.

Una vez que la nave haya divisado el núcleo, deberá dirigirse hacia él, y poniéndose lo más cerca posible, activar los tractos de arrastre, que posibilitarán que el núcleo se una a la nave a través de una barra rígida de Rwantita. A continuación, la nave deberá guiar el núcleo hasta la distancia de hiperespacio, considerándose la misión completa. El mayor peligro una vez cogido el núcleo, reside precisamente en éste, ya que su gran peso le impulsará hacia la superficie, pudiendo arrastrar asimismo la nave de la resistencia.

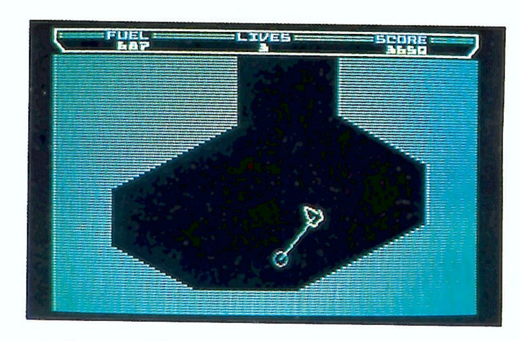
Pero por si no lo has adivinado ya, el elegido para la misión eres tú, así que ajústate el mono de astronauta y ¡Suerte!, (va a serte muy necesaria).

VEREDICTO FINAL

GRAFICOS	***
SONIDO	***
ORIGINALIDAD	***
DIFICULTAD	Progresiva
INTERES	*****







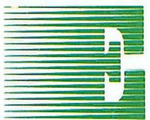
FICHA TECNICA

NOMBRE	THRUST
PRECIO	699 PTAS.
SOPORTE	CINTA
TIPO	ACCION
MODELO	C-64, C-128
OBSERVACIONES	SE MANEJA CON TECLADO



Ninja

¿Cómo es posible crear la más perfecta máquina de lucha?, esta debe ser inquebrantable, valiente, incansable y poderosa, precisa e invulnerable, lo cual es casi un sueño; pero largos años de entrenamiento continuado, vida austera, ausencia de placeres y concentración absoluta, le han acercado lo bastante a ese ideal. Ninja fue apartado de sus compañeros debido a sus cualidades innatas de arrojo y fuerza, y preparado concienzudamente en todas las artes marciales, llegando a ser cinturón negro en todas ellas, pero incluso para Ninja, la misión que se le ha encomendado parece gigantesca.


Experto dominador del karate, jiu-jitsu, lucha japonesa y demás artes orientales, Ninja ha sido además perfeccionado en el empleo de armas arrojadizas, tales como dagas o estrellas de la muerte, siendo además muy hábil en el uso de la katana o espada samurai. Todo ello, unido a sus puños de acero y a sus pies, precisos en sus ataques, le convierte en el candidato elegido para liberar a la princesa Di-Di, Perla de Oriente, del Palacio de las Gemas, donde se encuentra secuestrada por un enemigo real, y custodiada por legiones de samurais.

Ninja deberá ir desembarazándose de todos sus enemigos a través del palacio hasta encontrar a la princesa, cosa nada fácil. Parece claro que nuestro amigo llevará a cabo su misión casi sin despeinarse, gracias a los antecedentes que hemos dado de sus cualidades, pero no es así, ya que la guardia del Palacio está compuesta de Ninjas como él, igualmente entrenados, pero que se pasaron al otro bando, además de karatekas y thugs, todos peligrosos enemigos.

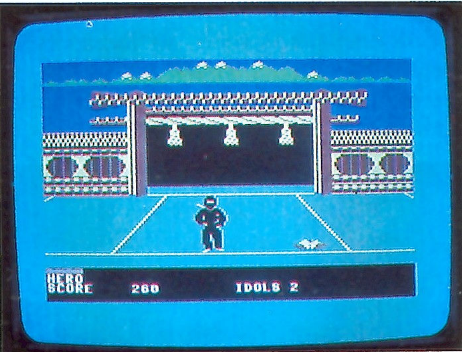
VEREDICTO FINAL

GRAFICOS	*****
SONIDO	****
ORIGINALIDAD	***
DIFICULTAD	*****
INTÉRES	*****

A pesar de ser muy valiente, Ninja es vulnerable, y caera bajo muchos golpes, pero tiene la oportunidad de recoger los ídolos que vaya encontrando, los cuales le pondrán de nuevo al máximo de fuerzas. Con todo, cada enemigo de Ninja, por separado y estando éste en plenitud de fuerzas no supondrá un obstáculo serio para él, pero lo malo es que en ocasiones sus oponentes aparecerán en un número de cuatro o cinco, desgastándole cada uno más. Los más peligrosos serán los Ninjas contrarios, que también lanzan dagas, estrellas y manejan endiabladamente la espada; los karatekas son rápidos y una patada suya es brutal, por último, los thugs son muy ágiles, a menos que se les encuentre en meditación trascendental, momento que será propicio para atacarles.

Con estas nociones debería bastarte para rescatar a la princesa, pero si no es así, lo sentimos por ti, ya que tu vida será corta.







FICHA TECNICA

NOMBRE	NINJA
PRECIO	1.100 PTAS.
SOPORTE	CINTA
TIPO	ARTES MARCIALES
MODELO	C-64, C-128
OBSERVACIONES	¿¡DONDE ESTA LA PRINCESA!?



Parallax

Lo que parecía un planeta amigable en una primera apreciación, se ha convertido de repente en una ratonera de la que te será muy difícil salir. Habiendo tomado tierra en un mundo artificial descubierto accidentalmente, los primeros contactos con sus habitantes han sido buenos. El planeta está dividido en cinco grandes zonas, Alpha, Beta, Gamma y Epsilon, pero casualmente hemos detectado una información vital traspasada entre las zonas por ordenador y su mensaje revela un inminente ataque a la Tierra. Ya no se puede echar marcha atrás, el planeta se revela como era verdaderamente y Parallax nos muestra sus garras.

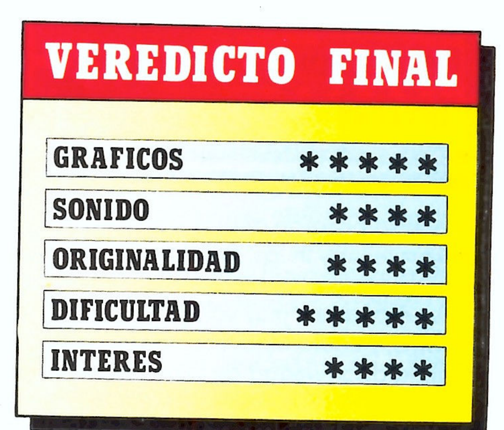


uestro propósito es ir pasando a través de las zonas del planeta, rescatando a nuestros compañeros, que han quedado diseminados por ellas, y llegar a la puerta intergaláctica, al final de la zona de Epsilon; asimismo, deberemos destruir el ordenador central.

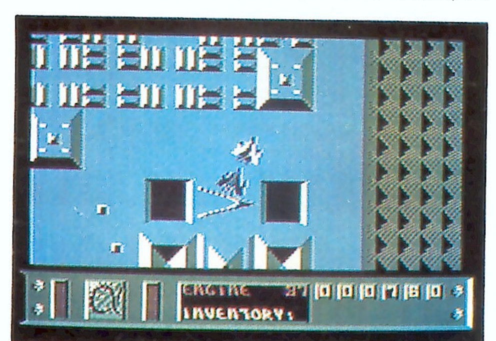
A bordo de Ibis, nuestra nave, poseemos equipos estándar de salida a la superficie, que tendremos que utilizar para entrar en los hangares existentes en cada nivel. En estos se encuentran varios ordenadores; de unos se podrán obtener créditos que habremos de utilizar en tiendas-ordenador, otros serán ficheros de tarjetas de datos, de los que deberemos extraer la información necesaria que nos permita la salida de cada zona. Por último las U.C.I. o unidades centrales de inteligencia son las que planean el ataque a la Tierra, y sólo serán desactivadas mediante determinadas claves que habremos de quitar a los científicos de cada zona, disparándoles para inmovilizarles.

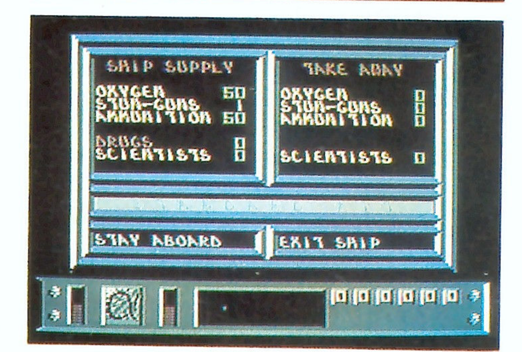
Con nuestra nave Ibis deberemos ir buscando los hangares a través de cada nivel, intentando aterrizar lo más cerca posible de éstos para que nuestras reservas de oxígeno en tierra no se agoten. También nos defenderemos de los alienígenas que nos atacan, los cuales pueden producirnos importantes daños en nuestro Ibis, puesto que aunque dispongamos del escudo protector de la nave, éste gasta mucho más combustible del normal cuando está conectado, y nuestras reservas no son imperecederas.

Aprieta bien todos los cabos que te lleven a la clave de cada nivel y escapa de Parallax.



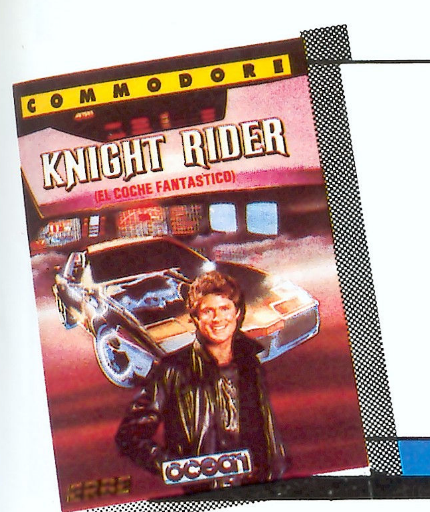
VEREDICTO FINAL	
GRAFICOS	* * * * *
SONIDO	* * * *
ORIGINALIDAD	* * * *
DIFICULTAD	* * * * *
INTERES	* * * *





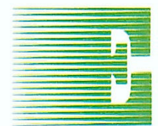
FICHA TECNICA

NOMBRE	PARALLAX
PRECIO	2.100 PTAS.
SOPORTE	CINTA
TIPO	ESTRATEGIA Y ACCION
MODELO	C-64, C-128
OBSERVACIONES	



Knight Rider

Michael, Devon quiere hablar contigo. De acuerdo, KITT, ponme con él —contestó Mike—. «Hola, Michael, me ha llegado información secreta acerca de la existencia de un grupo de terroristas que quieren alterar la paz entre las dos potencias, con el peligro de provocar una Tercera Guerra Mundial, descubre el plan y acaba con él o todos moriremos». A la orden, Devon. ¿Has oído, KITT?, pues en marcha.



n efecto, estás embarcado en una peligrosa aventura de la que sólo saldrás indemne realizando una perfecta labor de equipo con Devon, tu coordinador, y con KITT, el increíble prototipo de Industrias Knight. No conoces el plan de los terroristas, ni el tiempo que te queda para descubrirlo e interceptarlo, pero dispones de la información de unos importantes mapas que te señalan los puntos más probables de Norteamérica para llevarlo a cabo.

El juego se divide en tres fases, cada una de ellas importante por igual. La primera es la de mapas, donde deberás elegir una ruta lo más corta posible para llegar al punto donde Devon cree que puedes hallar la base terrorista. Una vez hallas seleccionado la ruta a seguir, deberás emplear toda tu pericia para llegar cuanto antes a ese punto en cuestión. Podrás elegir dos modalidades de conducción, una en la que tú, Michael Knight, controlas el coche, mientras que KITT ejerce la labor de disparar con el láser a los helicópteros que os atacan, o viceversa. KITT es más tranquilo que tú conduciendo, así que si prefieres ir a mayores velocidades, coge tú el volante, pero recuerda que cuanto más rápido vayas, más difícil le será a tu compañero controlar la precisión de sus disparos.

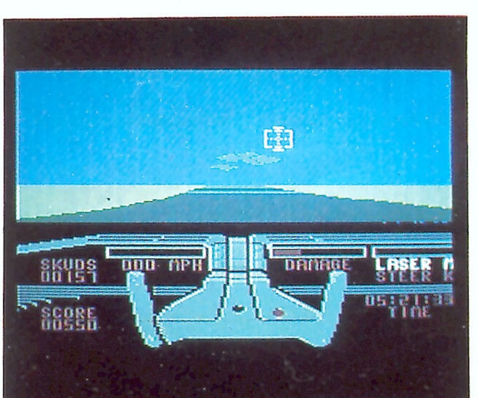
Todos los daños que os inflijan los helicópteros durante la travesía serán reparados instantáneamente al llegar a la ciudad objetivo. El láser tiene una utilidad mayor cuanto menos lo recalientes, puesto que al llegar a este punto dejará de funcionar, por ello deberás seleccionar cuidadosamente tus blancos y acertar a la primera, si tu puntería te lo permite.

La tercera fase tiene lugar en las bases terroristas, en las cuales Devon te dará instrucciones acerca de lo que debes hacer. Así que pisa el acelerador con entusiasmo, hasta el... final de tu aventura.

VEREDICTO FINAL

GRAFICOS	****
SONIDO	****
ORIGINALIDAD	*
DIFICULTAD	*****
INTERES	***





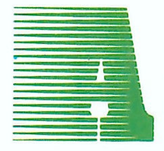
FICHA TECNICA

NOMBRE	KNIGHT RIDER
PRECIO	2.500 PTAS.
SOPORTE	CINTA
TIPO	ACCION
MODELO	C-64, C-128
OBSERVACIONES	¡NO TIENE TURBO BOOST!



Hole in One

Andaba el hombre primitivo con una tranca a sus hombros por la orilla del río, cuando empezó a pensar en lo aburrida que era la vida, sin ordenadores, ni viajes al Caribe, ni deportes. De repente vio un perfecto canto rodado y hastiado lo golpeó con fuerza ayudándose de su tranca; el canto se elevó a través de los árboles, yendo a caer en el agujero que un topo había labrado en el suelo, introduciéndose en su interior.



Asombrado por su suerte, exhaló una especie de gruñido gutural: «¡groulf!», y yendo a buscar la piedra, lo intentó de nuevo. Aunque lo cierto es que no consiguió su objetivo por segunda vez, enseñó a sus vecinos el juego, y éste se fue contando de padres a hijos hasta llegar a nuestros días, un poco más evolucionado. En homenaje a su inventor, que no tenía nombre, se le decidió llamar Golf, una derivación de la primera exclamación proferida por éste al descubrirlo.

El golf poco a poco fue abriéndose camino, y empezó a practicarse en el campo de hierba del vecino (de ahí que fuera de doble utilidad la tranca), hasta hoy día, cuando una organización perfecta respalda a un deporte como este. Los campos de golf, de una longitud opcional, están cubiertos de hierba, y rodeados de dificultades para sus practicantes, tales como árboles, riachuelos, bunkers (agujeros de arena), y las propias del terreno mismo.

Con el tiempo, este deporte ha llegado también al ordenador, y en este caso «Hole in One» es una buena adaptación. Proporciona algunas opciones importantes, tales como la elección de palos (madera, hierro, PW, SW, y putter), la dirección que se le quiere imprimir a la bola, así como la facultad de regular nosotros mismos el «swing» y por tanto, la fuerza del golpe. Otros alicientes serán la cambiante fuerza del viento, o las grandes dificultades que rodean a las «calles» antes de llegar al «green».

Tendremos un determinado «par de golpes» para cada hoyo, hasta un total de 18; el «par» de cada hoyo variará según sea su longitud, siendo los más pequeños de par 3 y los de gran longitud de par 5. De todas formas, no será posible cumplir ese número de golpes en casi ningún hoyo, debido a las dificultades del terreno; si opinas lo contrario, no tienes más que probar.



VEREDICTO FINAL

GRAFICOS	****
SONIDO	***
ORIGINALIDAD	**
DIFICULTAD	*****
INTERES	****

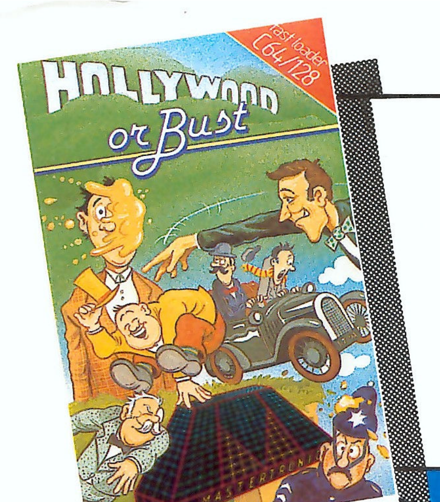






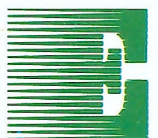
FICHA TECNICA

NOMBRE	HOLE IN ONE
PRECIO	750 PTAS.
SOPORTE	CINTA
TIPO	DEPORTES
MODELO	C-64, C-128
OBSERVACIONES	



Hollywood or Bust

Nos encontramos inmersos en el gran mundo del espectáculo, donde las sonrisas se reparten por doquier, unas veces verdaderas, otras simuladas, donde la competencia es feroz y un papel principal en una buena película equivale casi con seguridad a la gloria; multitud de filmes empiezan a rodarse cada día, con los guiones más dispares que se pueda imaginar, los decorados corren de un lado a otro, las pruebas se suceden, la vida transcurre a un ritmo vertiginoso, es lógico, esto es ¡Hollywood!



En este caso, nuestro guión indica que el protagonista, Buster Baloney, el cual ha logrado acceder al papel gracias a unos estupendos jamones de Jabugo que, curiosamente, han llegado a manos del director, debe en su papel buscar cinco Oscars que se han perdido en el estudio. Lo malo es que nadie sabe dónde se encuentran, así que el éxito de la película depende únicamente de la pericia de nuestro héroe en su búsqueda.

Además, y desafortunadamente para Buster, su sueldo también depende de su olfato personal para encontrar las estatuillas, por lo que en cuanto el director grita las consabidas palabras: ¡cámara!, ¡acción!, ¡se rueda!, nuestro amigo se lanza a su búsqueda frenéticamente.

Pero las trabas no han acabado para Buster, puesto que el estudio está custodiado por guardias de seguridad y otras criaturas, que le pondrán muy difícil la entrada en él; sólo burlándoles o lanzándoles pasteles de nata, de los que lleva una muy aparente reserva, logrará su objetivo. Lo más seguro es que su entrada movilice a toda la policía de Hollywood y se lancen en su persecución, pero ahí es donde la agilidad de Buster (producto de años de práctica) debe aparecer para driblarles en un callejón al compás de una graciosa música típica de las películas mudas, y si es posible quitarse de encima unos cuantos con sus tartitas de nata.



VEREDICTO FINAL

GRAFICOS	****
SONIDO	*****
ORIGINALIDAD	*****
DIFICULTAD	*****
INTERES	*****

Cada vez que Buster sea atrapado por la policía u otros sujetos, el director cortará muy enfadado la escena, lo cual irá en decremento del salario de nuestro actor preferido. Como que ya estás puesto en antecedentes, sé bueno y ayuda a Buster a encontrar los preciados Oscars.

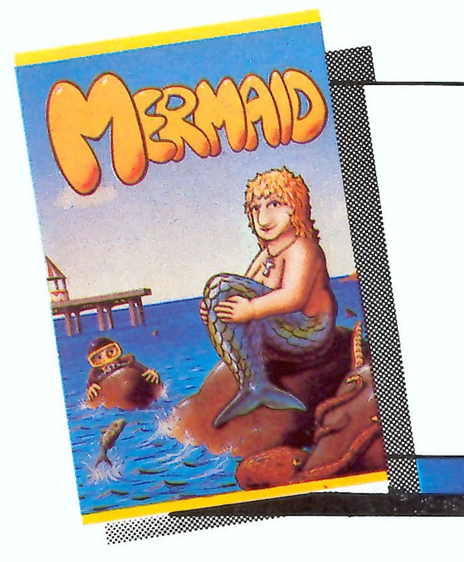






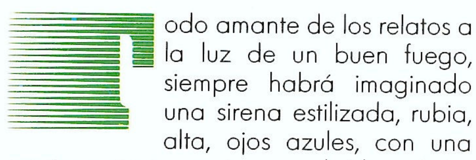
FICHA TECNICA

NOMBRE	HOLLYWOOD OR BUST
PRECIO	750 PTAS.
SOPORTE	CINTA
TIPO	ACCION
MODELO	C-64, C-128
OBSERVACIONES	¡QUE MUSICA!



Mermaid Madness

El mito de la sirena ha aparecido desde lejanos tiempos en las leyendas de los pueblos al borde del mar. Extrañas cualidades se les han atribuído, tales como el poder de atracción que tiene la voz de una de estas sílfides (Homero nos lo relata en «La Odisea»). Pero, ¿es realmente una sirena tal como nos la imaginamos?, apareciendo en ocasiones bajo el mar salvando a personas que estaban en peligro de muerte, yendo a varar a las orillas de un puerto en busca del rostro de marino, o surgiendo de entre las aguas y realizando durante unas milésimas de segundo un salto magistral, que hace creer a los tripulantes de un barco que han bebido más de la cuenta. Todas estas versiones han sido ya narradas, pero quien quiera saber algo más de las sirenas sólo tiene que entrar en el mundo de Mermaid Madness.



Todo amante de los relatos a la luz de un buen fuego, siempre habrá imaginado una sirena estilizada, rubia, alta, ojos azules, con una sonrisa entre inocente y encubridora de terribles secretos, en definitiva, un rostro de ensueño sobre un cuerpo perfecto (a no ser por esa cola de pescado).

Quizás lo mismo pensaba Gormless Gordon hasta que aquel día aciago de pesca submarina, se encaramó fatigosamente a los pilares del extremo del espigón, deseando no haber salido a la mar (no sabía lo que le esperaba). De pronto ante él, la aparición: una gigantesca sirena (que no sílfide), de 1,70 m y 100 kg de peso aproximadamente (a ojo de buen cubero). Nuestro amigo, ante semejante voluminosa sorpresa, fue vencido físicamente, y su cuerpo se precipitó medio desvanecido hacia las profundidades del mar, con tan mala suerte que quedó enganchado en el amasijo de hierros de los restos del naufragio de un barco.

Volviendo al momento en que sus ojos se cruzaron, Myrtle, que así se llamaba la sirena, sintió que el fuego del amor subía a través de su «estilizado» cuerpo, y al ver hundirse al objeto de su deseo, tomó carrerilla y lanzose al agua en su búsqueda, provocando poco menos que una tromba marina. De este modo, nuestra sirenita, al ver la situación en que se encontraba su amado, tomó la decisión de librarle de su enredada situación, evitando los peligros del mar, para llevar a buen término esta historia de amor y miedo al aplastamiento, respectivamente.





VEREDICTO FINAL

GRAFICOS	****
SONIDO	****
ORIGINALIDAD	*****
DIFICULTAD	*****
INTERES	***

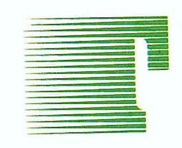
FICHA TECNICA

NOMBRE	MERMAID MADNESS
PRECIO	2.300 y 3.700 PTAS.
SOPORTE	CINTA Y DISCO
TIPO	ACCION
MODELO	C-64, C-128
OBSERVACIONES	



SUPER CYCLE

La bandera acaba de bajar y nos lanzamos a una velocidad de vértigo a través de la pista. El asfalto va desapareciendo ante nosotros como si fuera verdaderamente tragado por nuestra potente máquina, las manos empiezan a agarrotarse en las empuñaduras y los roces de nuestra rodilla en el suelo comienzan a ser cada vez más dolorosos; las curvas se suceden peligrosamente y nuestro cerebro va controlando con exactitud milimétrica cada viraje, cada punto de embrague, cada apurada frenada, puesto que la vida pende de un hilo a lomos de una 750 cc. en Super Cycle.



Todo el realismo de una carrera de motos está contenido en Super Cycle, donde se aprecia perfectamente cada movimiento del piloto en su montura. La carrera tiene lugar en diferentes lugares de todo el mundo y va aumentando progresivamente de dificultad, conforme seamos capaces de ir superando los distintos circuitos en que se nos somete a prueba.

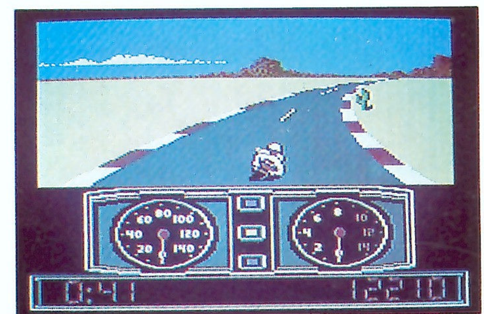
Los instrumentos nos informan en todo momento de la situación en que nos encontramos; el velocímetro, capaz de medir una velocidad de 140 millas/hora, y el contador de revoluciones, que pueden subir hasta 17000/minuto. El mismo sonido de la moto nos hará saber cuándo es propicio el cambio de marcha durante la carrera. Lo mejor es revolucionar la moto bastante antes de cambiar, de lo contrario en una marcha superior nuestra máquina irá muy ahogada y tardará más en alcanzar una buena velocidad. Disponemos de una caja de cambios de tres marchas, y veremos en nuestros instrumentos en cuál nos hallamos en cada momento.

En la salida competimos con otras dos motos similares a la nuestra, e iremos encontrando otras a medida que avancemos en la carrera. Tendremos un tiempo máximo para alcanzar la meta, por encima del cual no habremos superado el circuito. Estos irán subiendo progresivamente de dificultad, y aparecerán ante nosotros manchas de aceite, baches, e incluso vallas, con las cuales será muy difícil no colisionar.

Unicamente un piloto experto podrá conseguir llegar al final y proclamarse vencedor, ¿eres tú el elegido?







VEREDICTO FINAL

GRAFICOS	*****
SONIDO	****
ORIGINALIDAD	**
DIFICULTAD	Seleccionable
INTERES	*****

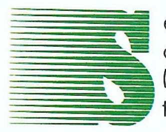
FICHA TECNICA

NOMBRE	SUPER CYCLE
PRECIO	2.300 PTAS.
SOPORTE	CINTA
TIPO	DEPORTES
MODELO	C-64, C-128
OBSERVACIONES	



Miami Vice

Todos hemos oído hablar del mundo sumergido del hampa, donde introducirse es excesivamente difícil y peligroso, que oculta bajo su impoluta fachada terribles secretos: tráfico de drogas, casuales «accidentes», desapariciones misteriosas, grandes operaciones con dinero ilegal, planes internacionales... Sus tentáculos alcanzan casi todos los estratos sociales, y los métodos con que se llevan a cabo sus planes tienen como factor común la rapidez y limpieza. Si quieres conocer este mundo de cerca, sólo tienes que acercarte a... Miami Vice.



Se rumorea en los ambientes conocidos por la policía (que también tiene gente introducida en las organizaciones criminales), que un gran alijo de droga (1 millón de dólares) va a ser introducido en Miami el jueves por la mañana, con destino a las manos del Sr. J., un aparente hombre de negocios, pero jefe de una gran red internacional del narcotráfico.

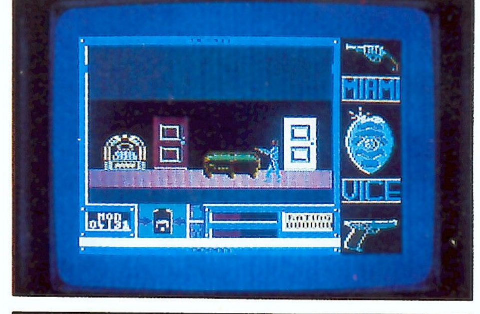
Con ese escape de información, el jefe de policía ha decidido interceptar el cargamento, para lo que envía a sus mejores hombres tras las huellas del envío de droga. De este modo, nosotros, Crockett y Tubbs, en estrecha colaboración, deberemos recorrer la ciudad en nuestro coche, visitando en primer lugar a los camellos más frecuentes, los cuales con suerte nos harán llegar hasta sus proveedores, y estos a los jefes de casinos y salas de juego, siempre mucho más cerca del Sr. J.

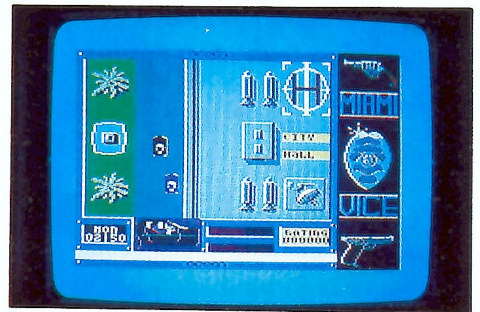
Conocemos determinados sitios que se han filtrado a través de la casi perfecta organización de este gangster, donde se dan cita algunos maleantes de la ciudad, conociendo asimismo la hora de reunión, así que tendremos que presentarnos en esos lugares de improviso y sonsacar todo lo que podamos a los allí presentes. Siempre que encontremos pruebas (bolsas de heroína) o arrestemos sospechosos, podremos llevarlos al Ayuntamiento, desde donde serán puestos a buen recaudo.

Sólo conseguiremos acabar con nuestra misión una vez hallamos encontrado al Sr. J., pero eso nos costará mucho trabajo, interminables visitas a bares y casinos, grandes sobornos y peligrosas persecuciones por la ciudad; pero recuerda que es inevitable, ya que estás inmerso en Miami Vice... ¿saldrás indemne?

VEREDICTO FINAL

GRAFICOS	****
SONIDO	****
ORIGINALIDAD	*****
DIFICULTAD	*****
INTERES	***



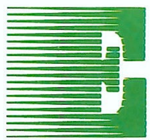


FICHA TECNICA

NOMBRE	MIAMI VICE
PRECIO	2.500 PTAS.
SOPORTE	CINTA
TIPO	ACCION Y ESTRATEGIA
MODELO	C-64, C-128
OBSERVACIONES	

Interrupciones en la práctica

Vamos a ver en esta ocasión un par de ejemplos sobre el empleo de las IRQ en nuestro Commodore. Como siempre, pretendemos realizar ejemplos útiles, que puedan ser empleados directamente, o que sirvan de punto de partida para la experimentación.



En el número de noviembre hemos visto detalladamente qué son las IRQ, su función y cómo se emplean. Vamos a realizar a continuación dos ejemplos prácticos sobre este tema.

Los ejemplos se han diseñado en un C-64, pero con algunas modificaciones, el programa «teclado sonoro» puede funcionar en cualquier equipo Commodore; el ejemplo «movimiento sprites» sólo se puede introducir en un 64 ó 128, por ser los únicos miembros de la familia commodore que se hallan dotados de esta facilidad gráfica.

Como siempre, los ejemplos se presentan en un listado en ensamblador, para que aquellos que dispongáis de uno podáis introducirlos directamente, modificarlos y experimentar con ellos. Para aquellos que no tengáis acceso a un ensamblador, se acompañan unos listados en BASIC que permiten introducir en memoria los programas.

Y tras la introducción, vayamos al grano...

Teclado, sonido e interrupciones, todo unido en un programa

Nuestro primer ejemplo será la realización de un teclado sonoro, lo que algunos especialistas llaman un «feedback» acústico (con perdón).

Básicamente, nuestra meta es desarrollar un programa que detecte si se está pulsando alguna tecla; si no, no debe ocurrir nada, pero si se oprime alguna, ha de producirse un sonido: no el clásico «crack» de cuando se rompe una tecla, sino uno más sutil y, por supuesto más inofensivo para la salud de nuestro equipo.

El programa nos servirá para confirmar si se ha pulsado correctamente la tecla correspondiente, sin necesidad de observar la pantalla.

Para sofisticar un poco el proyecto, y así poder pasmar a los vecinos, nuestro programa también diferenciará si se ha pulsado la tecla SHIFT, CTRL o COMMODORE, y producirá un sonido más agudo si se ha presionado alguna de éstas.

El programa se puede observar en los primeros listados, tanto en ensamblador como en su versión BASIC lista para introducir. La primera parte es la que se ejecuta directamente, y únicamente modifica el vector IRQ, en las posiciones 788 y 789 para que apunte al inicio de nuestra rutina.

Notemos que lo primero que se hace es inhibir las interrupciones. ¿Por qué? Si se produce una IRQ en el crítico instante en que se ha cambiado el primer byte del vector, pero no el segundo, el microprocesador se encontrará con un vector erróneo, mitad el original, mitad el nuevo, que no apunta al lugar correcto; de ser así, el salto se realizaría a un lugar indeterminado, con las catastróficas consecuencias fáciles de imaginar. Para evitar esta posibilidad, se inhiben las interrupciones mientras se cambia el vector.

Una vez modificado, se restauran las interrupciones y se vuelve al BASIC, con la instrucción RTS. Nuestra rutina se ejecutará ahora 50 veces por segundo, sin necesidad de que la llamemos. Para activarla, hay que utilizar SYS 49152 (o JSR $C000, en ensamblador) y para desactivarla, la secuencia RUN-STOP/RESTORE.

Cómo utilizar algunas direcciones estratégicas del sistema

El resto de la rutina es muy sencilla. En primer lugar, explora el contenido de la posición de memoria 197, que contiene un código determinado, según la tecla pulsada. Si no se ha pulsado ninguna, esta posición de memoria contiene un 64.

La estructura es obvia: si hay un 64, no se está pulsando ninguna tecla, y, por tanto, se debe apagar el sonido que pudiese existir (sitúa un cero en el control de volumen, posición 54296).

Si existe un valor distinto de 64, se pasa a la sección del programa que decide si se ha pulsado la tecla SHIFT. Esto se realiza explorando el contenido de la posición de memoria 653, que contiene un cero si no se pulsa SHIFT, CTRL o COMMODORE.

En caso de haber pulsado SHIFT, el bit 0 de esta posición se pone a uno; si se pulsa COMMODORE, se activa el bit 1, y el 2 si se oprime CTRL. Por tanto, se obtienen valores diferentes según se pulsen estas teclas, pudiendo detectarse la pulsación simultánea de dos de ellas, e incluso de las tres.

Si existe un dato distinto de cero, se carga un valor determinado para la frecuencia de la nota a emitir. Si es cero, se carga un valor más bajo para la frecuencia, dando un tono más grave.

El resto del programa simplemente coloca los valores adecuados en los registros del SID, para generar el sonido: forma de onda, valor ADSR y, finalmente, el volumen.

Por último, se salta a la rutina del sistema operativo de Commodore, la cual se halla en la dirección $EA31, y realiza las funciones más necesarias, como lectura de teclado, actualización del reloj del sistema, etc.

Si no accediésemos a esta rutina tras realizar nuestras funciones, el sistema quedaría bloqueado, al no poder ni siquiera leer el teclado.

Hasta aquí, queda explicado el funcionamiento de la rutina. Un análisis detenido del listado en ensamblador puede aclarar las posibles dudas sobre la misma. Observemos que en dicho listado se emplean asíduamente las etiquetas; el trabajo de programación se simplifica así notablemente, e incluso el listado del programa es más fácil de leer y comprender.

Desgraciadamente, no todos los ensambladores disponen de esta característica. Si

el vuestro no la tiene, simplemente sustituid las etiquetas por los valores de las direcciones correctas... O comprad un buen ensamblador, lo cual resultará aconsejable si deseáis realizar cualquier clase de programación mínimamente serio en código máquina.

Moviendo sprites por arte de magia... y del hechizo IRQ

Nuestro segundo ejemplo es algo más espectacular... y más largo, como podréis comprobar por la longitud del listado. Se trata de una rutina la cual permite que los sprites se muevan libremente por la pantalla, a una velocidad y en una dirección determinadas, sin que nosotros a través de nuestro programa, tengamos que tomar parte en ello.

Este tipo de rutina puede resultar de gran utilidad en multitud de juegos, donde mover simultáneamente varios sprites a velocidad razonable puede ser toda una aventura.

El listado en ensamblador se halla profusamente comentado, indicando prácticamente la función de cada línea. Por tanto, vamos a tratar someramente el funcionamiento de la misma.

Una primera parte realiza la ya conocida función de cambiar el vector IRQ. Prácticamente es igual al ejemplo anterior. A continuación, se inicializa un bucle; la siguiente parte del programa se repetirá 8 veces, una para cada sprite.

En primer lugar, se estudia el movimiento del sprite en el sentido x, es decir, de derecha a izquierda de la pantalla. La velocidad del sprite se controla mediante un contador, que es analizado cincuenta veces por segundo.

Cómo se inventó la velocidad...

El mencionado contador se inicializa a un valor relacionado con la velocidad del sprite: a alta velocidad, se carga un valor bajo en el contador, y viceversa.

El contador se decrementa cada vez que la rutina es ejecutada; si es nulo, el sprite se mueve, y si no lo es, a otra cosa...

De este modo, tal como hemos inicializado el valor del contador, con un valor bajo el sprite se mueve con más frecuencia, dando una velocidad mayor. Y al revés, un contador alto consigue que el programa tenga que decrementarlo más veces para hacerlo llegar a cero, moviéndolo menos (menor velocidad). El contador vuelve a ser inicializado cada vez que el sprite se mueve, repitiendo el proceso indefinidamente.

El valor de la velocidad se halla en el rango 0-31, siendo 31 la velocidad máxima, y disminuyendo con este valor, hasta llegar al 0, que inhibe el movimiento del sprite. El sentido del desplazamiento se controla por el bit 7 de la velocidad. Si este bit se halla activado (velocidad>128), el movimiento se realiza en sentido contrario (hacia la izquierda, en este caso). Valores de velocidad que difieran en 128 dan, por tanto, velocidad igual y contraria. Es decir, los valores 5 y 133 dan velocidades iguales, aunque de sentidos opuestos.

Por tanto, valores de la velocidad 0 y 128 hacen que el sprite se quede fijo en la pantalla.

Las siguientes líneas del programa simplemente controlan el proceso descrito hasta ahora; deciden si se mueve o no el sprite, actualizan el contador convenientemente y modifican la posición X del sprite, aumentando o disminuyendo el registro del chip VIC, según el sentido del movimiento.

Un problema adicional: manejar el bit más significativo...

Esto es suficiente en principio; pero la posición X de los sprites puede llegar a 320, valor que se sale del alcance de un byte. Para controlar esto, se debe poner a uno el bit adecuado del registro MSB del VIC.

Por ello, existe una parte del programa para llevar este control automáticamente. Si al aumentar el valor de X se produce un acarreo, se pone el bit a uno; o bien, si se produce al disminuir X un «no-acarreo» (recordad la operación de resta con acarreo), se pone a cero el bit correspondiente.

Para esto, se ha dispuesto la rutina MSB. Mediante ella, se cambia el bit correspondiente del registro MSB; mediante la operación EOR (OR-exclusivo), se pone a uno si se hallaba a cero, y viceversa.

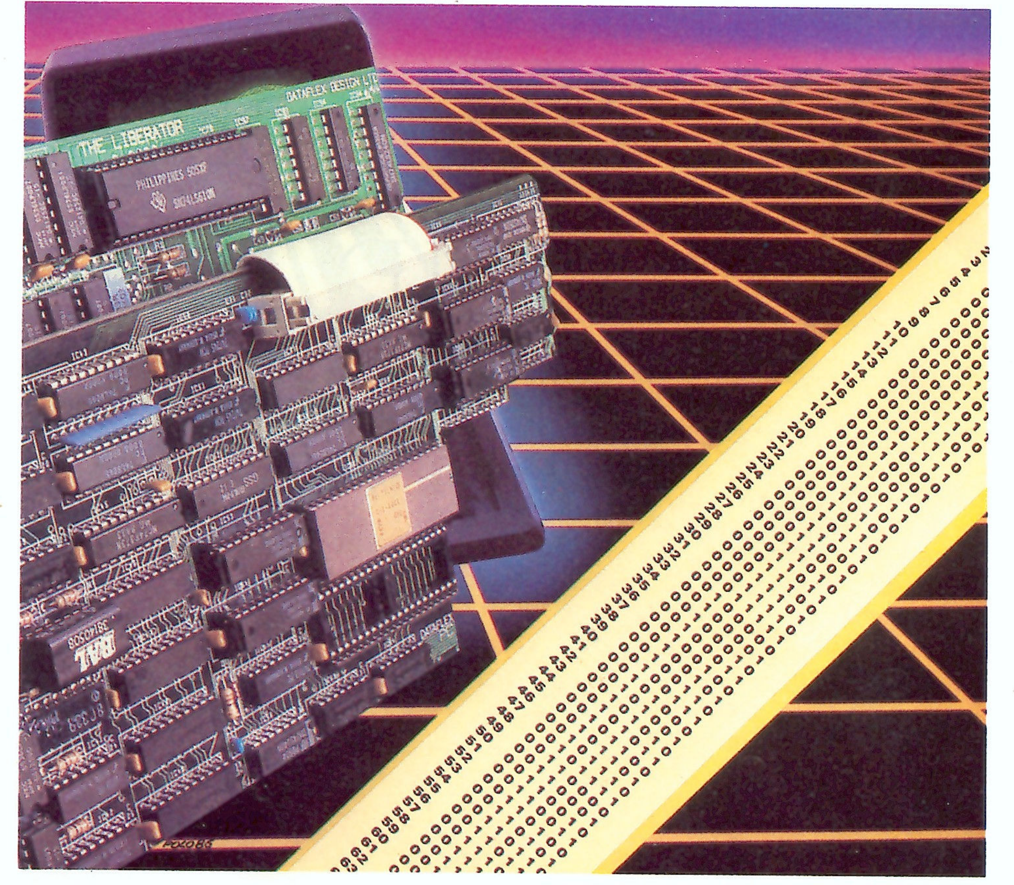
El resto es muy sencillo. Simplemente, se analiza el movimiento según el eje Y; esto se hace exactamente igual, aunque con la simplificación de no tener que ocuparnos del manejo del MSB.

Después, simplemente se repite el proceso, hasta que se hayan procesado los ocho sprites. Para abandonar la rutina, se salta a la dirección $EA31, donde el sistema se ocupa de las «tareas de la casa».

Las tablas: ahorro de tiempo y «materia gris», gasto de memoria

Al final del listado, podéis observar una serie de tablas, donde se almacenan datos de interés para el programa. En CONT se almacenan los contadores: dos para cada sprite, uno de velocidad horizontal y otro de velocidad vertical.

En la tabla MASCARA se tienen los valo-



res de los bits para cada sprite; accediendo a esta con una dirección por registro X (LDA MASCARA,X), incluyendo en este el número del sprite (0-7) se obtiene el valor del bit para el registro MSB. También podría realizarse esta tarea mediante una serie de operaciones lógicas (AND-LSR), pero de este modo ganamos en simplicidad y en rapidez, a costa de sólo 16 bytes adicionales de memoria.

Finalmente, la última tabla contiene los valores de las velocidades del sprite. Comienza a partir de la dirección 49390, y a ella debemos acceder para asignar los valores de velocidad deseados a cada sprite. Las velocidades se hallan en orden: para sprite 1, velocidad x y velocidad y; para sprite 2, velocidad x y velocidad y, etc.

Mediante sentencias POKE (o STA, si empleamos el código máquina) dejaremos en estas posiciones los valores que deseemos para las velocidades de los sprites. Activando la rutina con SYS 49152, los sprites se moverán por la pantalla... siempre y cuando hayan sido activados y ubicado previamente en una posición visible.

Con estos ejemplos, esperamos haber aclarado el empleo de las IRQ. Al menos de las IRQ de reloj, pues de las interrupciones generadas por el VIC nos ocuparemos próximamente, haciendo antes una breve pausa para describir al chip que tantas buenas imágenes nos regala: el chip VIC II.

LISTADO

```
--------------------------------------------------------------------------
A LISTING, PAGE #1
--------------------------------------------------------------------------


PASS 1

PASS 2

LINE# LOC. OBJECT LABELS           LINE

   10 C000                         .ORG    49152        ;ORIGEN DEL PROGRAMA
   20 C000 78                      SEI                  ;INHIBE IRQ
   30 C001 A90D                    LDA #INICIO&255      ;CAMBIA VECTOR POR
   40 C003 8D1403                  STA 788              ;EL DE NUESTRA RUTINA
   50 C006 A9C0                    LDA #INICIO/256
   60 C008 8D1503                  STA 789
   70 C00B 58                      CLI                  ;RESTAURA IRQ
   80 C00C 60                      RTS                  ;VUELVE AL BASIC
   90 C00D A5C5   INICIO           LDA 197              ;COMPRUEBA SI EXISTE ALGUNA
  100 C00F C940                    CMP #64              ;TECLA PULSADA
  110 C011 D008                    BNE TECLA
  120 C013 A900   FINAL            LDA #0               ;SI NO, APAGA SONIDO
  130 C015 8D18D4                  STA 54296
  140 C018 4C31EA                  JMP $EA31            ;SALTA A LA IRQ DEL SISTEMA
  150 C01B AD8D02 TECLA            LDA 653              ;MIRA SI SE PULSA SHIFT
  160 C01E D005                    BNE SHIFT
  170 C020 A921                    LDA #33              ;SI NO, CARGA TONO GRAVE
  180 C022 4C27C0                  JMP SONIDO
  190 C025 A93C   SHIFT            LDA #60              ;SI, CARGA TONO AGUDO
  200 C027 8D00D4 SONIDO           STA 54272            ;ALMACENA TONO EN REGISTRO
  210 C02A 8D01D4                  STA 54273            ;DE FRECUENCIA DEL SID
  220 C02D A980                    LDA #128             ;ATAQUE/CAIDA
  230 C02F 8D05D4                  STA 54277
  240 C032 A980                    LDA #128             ;SOSTENIMIENTO/RELAJACION
  250 C034 8D06D4                  STA 54278
  260 C037 A911                    LDA #17              ;FORMA DE ONDA
  270 C039 8D04D4                  STA 54276
  280 C03C A909                    LDA #9               ;VOLUMEN
  290 C03E 8D18D4                  STA 54296
  300 C041 4C31EA                  JMP $EA31            ;FIN, SALTA A IRQ SISTEMA

SUCCESSFUL ASSEMBLY; NO ERRORS.
```

```
--------------------------------------------------------------------------
A LISTING, PAGE #1
--------------------------------------------------------------------------


PASS 1

PASS 2

LINE# LOC. OBJECT LABELS           LINE

   10 C000                         .ORG    49152        ;ORIGEN DEL PROGRAMA
   20 C000          V              =$D000               ;VALOR DE LA VARIABLE V
   30 C000 78                      SEI                  ;INHIBE IRQ
   40 C001 A90D                    LDA #INICIO&255      ;CARGA EN NUEVO VECTOR
   50 C003 A2C0                    LDX #INICIO/256      ;DE LA RUTINA IRQ, Y LO
   60 C005 8D1403                  STA 788              ;SITUA EN MEMORIA
   70 C008 8E1503                  STX 789
   80 C00B 58                      CLI                  ;RESTAURA IRQ
   90 C00C 60                      RTS                  ;VUELVE AL BASIC
  100 C00D A200   INICIO           LDX #0               ;INICIALIZA BUCLE PRINCIPAL
  110 C00F A000                    LDY #0
  120 C011 BDDEC0 LOOP             LDA CONT,X           ;CARGA CONTADOR MOVIMIENTO
 X
  130 C014 F003                    BEQ *+5              ;SI NULO, MUEVE SPRITE
  140 C016 DEDEC0                  DEC CONT,X           ;NO, DECREMENTA Y COMPRUEBA
  150 C019 D039                    BNE CALY             ;SI NO NULO, NO MUEVE EN X
```

```
160 C01B BDEEC0          LDA VELX,X        ;CARGA VELOCIDAD
170 C01E 291F            AND #31           ;SOLO BITS 0-5, RESTO CERO
180 C020 F032            BEQ CALY          ;SI NULO, NO MUEVE EN X
190 C022 85FE            STA $FE           ;ALMACENA
200 C024 A91F            LDA #31           ;CONTADOR=31-VELOCIDAD
210 C026 38              SEC
220 C027 E5FE            SBC 254
230 C029 D002            BNE *+4           ;NULO, 1 EN CONTADOR
240 C02B A901            LDA #1
250 C02D 9DDEC0          STA CONT,X
260 C030 BDEEC0          LDA VELX,X        ;SI VELOCIDAD>128, A IZQDA
270 C033 2980            AND #128          ;RESTANDO UNO
280 C035 D035            BNE RESTA
290 C037 A901   SUMAR    LDA #1            ;SUMA 1 A LA X DEL SPRITE
300 C039 18              CLC               ;MOVIENDOLO A LA DERECHA
310 C03A 7D00D0          ADC V,X
320 C03D 9D00D0          STA V,X
330 C040 9003            BCC *+5           ;SI ACARREO, MODIFICA MSB
340 C042 2062C0          JSR MSB
350 C045 BDCEC0          LDA MASCARA,X     ;SEGUN SPRITE, BIT
360 C048 2D10D0          AND V+16          ;SI MSB=0, MOVIMIENTO Y
370 C04B F007            BEQ CALY
380 C04D BD00D0          LDA V,X           ;SI MSB=1, MIRA SI SALE
390 C050 C978            CMP #120          ;DE PANTALLA(X>376)
400 C052 1003            BPL *+5
410 C054 4C93C0 CALY     JMP MOVY
420 C057 A900            LDA #0            ;SI, PON X=CERO
430 C059 9D00D0          STA V,X
440 C05C 2062C0 EN1      JSR MSB           ;MSB=0
450 C05F 18              CLC               ;SALTA CALCULO DE MOVIENTO
460 C060 90F2            BCC CALY          ;SEGUN EJE Y
470 C062 BDCEC0 MSB      LDA MASCARA,X     ;PONE MSB=1 SI MSB=0
480 C065 4D10D0          EOR V+16          ;Y VICEVERSA
490 C068 8D10D0          STA V+16
500 C06B 60              RTS
510 C06C BD00D0 RESTA    LDA V,X           ;RESTA 1 A SPRITE, MOVIENDO
520 C06F 38              SEC               ;A LA IZQDA
530 C070 E901            SBC #1

--------------------------------------------------------------------------------
A LISTING, PAGE #2
--------------------------------------------------------------------------------

LINE# LOC. OBJECT LABELS       LINE

540 C072 9D00D0          STA V,X
550 C075 B0DD            BCS CALY          ;MANEJO DEL MSB, COMO
560 C077 2062C0          JSR MSB           ;EN LA SUMA
570 C07A BDCEC0          LDA MASCARA,X
580 C07D 2D10D0          AND V+16          ;SI X<0, SE HACE X=370
590 C080 F0D2            BEQ CALY          ;PARA QUE EL SPRITE SALGA
600 C082 A978            LDA #120          ;POR LA DERECHA
610 C084 9D00D0          STA V,X
620 C087 BDCEC0          LDA MASCARA,X
630 C08A 0D10D0          ORA V+16
640 C08D 8D10D0          STA V+16
650 C090 4C54C0          JMP CALY
660 C093 E8     MOVY     INX               ;MOVIMIENTO SEGUN EJE Y
670 C094 BDDEC0          LDA CONT,X        ;CARGA CONTADOR, SI NULO MU
EVE
680 C097 F003            BEQ *+5
690 C099 DEDEC0          DEC CONT,X        ;SI NO NULO, DECREMENTA Y
700 C09C D025            BNE OTROSPRITE    ;MIRA SI NULO
710 C09E BDEEC0          LDA VELX,X        ;CALCULO DE CONTADOR
720 C0A1 291F            AND #31           ;COMO ANTES
730 C0A3 F01E            BEQ OTROSPRITE    ;SI VEL=0, NO MUEVE
740 C0A5 85FE            STA 254
750 C0A7 A91F            LDA #31
760 C0A9 38              SEC
770 C0AA E5FE            SBC 254
780 C0AC D002            BNE *+4
790 C0AE A901            LDA #1
800 C0B0 9DDEC0          STA CONT,X
810 C0B3 BDEEC0          LDA VELX,X        ;SI VELOCIDAD>127,
820 C0B6 2980            AND #128          ;RESTA 1 (ARRIBA)
830 C0B8 D006            BNE RES
840 C0BA FE00D0 SUM      INC V,X           ;SUMA 1 A Y (ABAJO)
850 C0BD 4CC3C0          JMP OTROSPRITE
860 C0C0 DE00D0 RES      DEC V,X           ;RESTA 1 A Y (ABAJO)
870 C0C3 E8     OTROSPRITE INX             ;SI YA VAN 8 SPRITES,
880 C0C4 E010            CPX #16           ;SALTA A IRQ DEL SISTEMA
890 C0C6 F003            BEQ *+5
900 C0C8 4C11C0          JMP LOOP          ;SI NO, REPITE
910 C0CB 4C31EA          JMP $EA31
920 C0CE                 ;TABLAS DE DATOS: VELOCIDADES,
930 C0CE                 ;CONTADORES Y MASCARA DE BITS
940 C0CE 01     MASCARA  .BYTE  1,1,2,2,4,4,8,8
940 C0CF 01
940 C0D0 02
940 C0D1 02
940 C0D2 04
940 C0D3 04
940 C0D4 08
940 C0D5 08
950 C0D6 10              .BYTE  16,16,32,32,64,64,128,128
950 C0D7 10
950 C0D8 20
950 C0D9 20
950 C0DA 40
950 C0DB 40
950 C0DC 80
950 C0DD 80
960 C0DE 01     CONT     .BYTE  1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1
960 C0DF 01
```

```
----------------------------------------------------------------------------
A LISTING, PAGE #3
----------------------------------------------------------------------------

LINE# LOC. OBJECT LABELS          LINE

  960 C0E0 01
  960 C0E1 01
  960 C0E2 01
  960 C0E3 01
  960 C0E4 01
  960 C0E5 01
  960 C0E6 01
  960 C0E7 01
  960 C0E8 01
  960 C0E9 01
  960 C0EA 01
  960 C0EB 01
  960 C0EC 01
  960 C0ED 01
  970 C0EE 80     VELX             .BYTE   128,128,128,128,128,128,128,128
  970 C0EF 80
  970 C0F0 80
  970 C0F1 80
  970 C0F2 80
  970 C0F3 80
  970 C0F4 80
  970 C0F5 80
  980 C0F6 80                      .BYTE   128,128,128,128,128,128,128,128
  980 C0F7 80
  980 C0F8 80
  980 C0F9 80
  980 C0FA 80
  980 C0FB 80
  980 C0FC 80
  980 C0FD 80

SUCCESSFUL ASSEMBLY; NO ERRORS.
```

```
10 REM ******************************               -071-
20 REM * LISTADO 1: TECLADO SONORO *                -209-
30 REM ******************************               -073-
40 S=0:FORI=49152TO49219:READA:S=S+A:POKEI,A:NEXT
                                                    -084-
50 IF S<>7666 THEN PRINT"ERROR EN DATAS!":STOP
                                                    -133-
60 DATA 120,169,13,141,20,3,169,192,141,21,3,88,96,16
5,197,201                                           -045-
70 DATA 64,208,8,169,0,141,24,212,76,49,234,173,141,2
,208,5                                              -151-
80 DATA 169,33,76,39,192,169,60,141,0,212,141,1,212,1
69,128,141                                          -095-
90 DATA 5,212,169,128,141,6,212,169,17,141,4,212,169,
9,141,24                                            -249-
100 DATA 212,76,49,234                              -119-

READY.                                              -173-
```

```
10 REM *********************************            -197-
20 REM *LISTADO 2: MOVIMIENTO SPRITES *             -007-
30 REM *********************************            -199-
40 S=0:FORI=49152TO49405:READA:S=S+A:POKEI,A:NEXT
                                                    -081-
50 IF S<>30136THENPRINT"ERROR EN DATAS!":STOP
                                                    -105-
60 DATA 120,169,13,162,192,141,20,3,142,21,3,88,96,16
2,0,160                                             -183-
70 DATA 0,189,222,192,240,3,222,222,192,208,57,189,23
8,192,41,31                                         -148-
80 DATA 240,50,133,254,169,31,56,229,254,208,2,169,1,
157,222,192                                         -150-
90 DATA 189,238,192,41,128,208,53,169,1,24,125,0,208,
157,0,208                                           -052-
100 DATA 144,3,32,98,192,189,206,192,45,16,208,240,7,
189,0,208                                           -054-
110 DATA 201,120,16,3,76,147,192,169,0,157,0,208,32,9
8,192,24                                            -244-
120 DATA 144,242,189,206,192,77,16,208,141,16,208,96,
189,0,208,56                                        -212-
130 DATA 233,1,157,0,208,176,221,32,98,192,189,206,19
2,45,16,208                                         -151-
140 DATA 240,210,169,120,157,0,208,189,206,192,13,16,
208,141,16,208                                      -024-
150 DATA 76,84,192,232,189,222,192,240,3,222,222,192,
208,37,189,238                                      -056-
160 DATA 192,41,31,240,30,133,254,169,31,56,229,254,2
08,2,169,1                                          -089-
170 DATA 157,222,192,189,238,192,41,128,208,6,254,0,2
08,76,195,192                                       -013-
180 DATA 222,0,208,232,224,16,240,3,76,17,192,76,49,2
34,1,1                                              -140-
190 DATA 2,2,4,4,8,8,16,16,32,32,64,64,128,128,1,1
                                                    -200-
200 DATA 1,1,1,1,1,1,1,1,1,1,1,1,1,1,128,128
                                                    -100-
210 DATA 128,128,128,128,128,128,128,128,128,128,128,
128,128,128                                         -163-

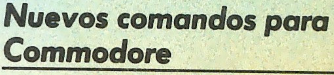
READY.                                              -173-
```

Text

He aquí una utilidad gigante a nuestra disposición con un pequeño esfuerzo: textos a diestro y siniestro, de pie, cabeza abajo y del tamaño que queramos.

La entrada de este artículo ya habrá sido lo suficientemente explicativa: nos encontramos ante una rutina de caracteres gigantes, que nos permitirá obtener una presentación de textos en pantallas de alta resolución, auténticamente fantástica. No sólo posibilita la edición de mensajes del tamaño que queramos (¡dentro de un orden! ¿eh?), sino que además podemos fijar la dirección y sentido en que deseamos que se escriban y otras características.

Visto que la utilidad del programa no se nos ocultan a nadie, pasemos directamente a estudiar la forma de emplearlo.



Nuevos comandos para Commodore

La aplicación se sintetiza a través de la implementación de seis nuevas instrucciones al lenguaje BASIC de Commodore.

@ HRS T,F,B: Prepara la pantalla de alta resolución, indicándose en el parámetro T el color de la tinta y en F el del fondo. B contendrá los valores 0 ó 1, según queramos un borrado previo o no de la pantalla HI-RES.

@ NRM: Carece de parámetros y permite el retorno a la pantalla de texto normal.

@ PLT X,Y,B: Se trata de una instrucción auxiliar, que permite dibujar un punto en el lugar determinado por las coordenadas X,Y. Nuevamente, si B vale cero el punto será desconectado, y si contiene un 1, encendido.

@ TXT X,Y,A,L,S,D,V,«...»: Esta es la instrucción maestra, dado que posibilita la escritura propiamente dicha, de cualquier texto situado entre comillas al final del comando, o bien en forma de expresión de cadena, siempre y cuando los códigos que contenga se encuentren entre 0 y 63.

La escritura se efectuará a partir del punto de coordenadas X,Y, con un ancho definido por el parámetro A y una longitud establecida mediante L. Por otra parte, S proporciona la separación entre los caracteres, y D la dirección y sentido de escritura, según la siguiente norma: 1=izquierda a derecha, 3=derecha a izquierda, 2=arriba a abajo y 4=abajo a arriba.
Por último, el parámetro V señalará una impresión en vídeo normal (0) o inverso (1).

@ DFC C,B1,B2,B3,B4,B5,B6,B7,B8: Sirve para la configuración de un carácter C, acorde con los datos proporcionados por los parámetros restantes.

@ SET P: Este último comando sitúa el inicio del generador de caracteres, indicando el byte alto (el bajo se supone 0) de la dirección donde se emplaza el primer byte del carácter de código cero.

Así por ejemplo, si a un programa que vaya a funcionar con la rutina, le añadimos las líneas:

POKE50209,216:POKE50211,204:POKE
50231,218:SYS49152
POKE50209,208:POKE50211,206:POKE
50231,210

y escribimos @ SET 206, representaremos las letras mayúsculas (posición normal) y con @ SET 204, minúsculas.

Ultimos consejos de uso

A título general, hemos de advertir que para los parámetros pueden ser utilizadas expresiones que den como resultado un número entero positivo y que tras una sentencia THEN será necesario preceder los nuevos comandos de dos puntos (:).

Por otra parte, la pantalla de alta resolución se ubica por debajo de la ROM del BASIC; la memoria de color a partir de $8000 y las rutinas de TEXT desde $C000. Por ello, al escribir en alta resolución, se puede producir una caída del sistema si rebasamos los topes superior o inferior de la pantalla, no así por los lados.

Asimismo, es aconsejable limitar el espacio disponible para programas y variables, con el fin que estos no interfieran la memoria de color. Un buen ejemplo de esta acción, lo tenemos en la línea 10 del programa de demostración que acompaña esta rutina.

Por último es de destacar que la secuencia RUN STOP/RESTORE no desconecta las nuevas instrucciones, siendo necesario recurrir a un RESET o SYS 64738. La reconexión se consigue mediante SYS 49152.

LISTADO

PROGRAMA DE DEMOSTRACION

```
10 POKE52,48:POKE56,48:POKE53280,1:POKE53281,1:@DFC0,
0,24,0,24,48,96,102,60                               -131-
20 POKE50209,216:POKE50211,204:POKE50231,218:SYS49152
                                                     -003-
30 POKE50209,208:POKE50211,206:POKE50231,210
                                                     -193-
40 @HRS6,1,0:@TXT10,92,2,2,16,1,0,"@QUE HORA ES? 00:0
0":X=234                                             -120-
50 GETA$:IFA$=""THENC=ABS(C-1):@TXTX,92,2,2,0,1,C,"0"
:GOTO50                                              -181-
60 IFA$<"0"ORA$>"9"THEN50                            -127-
70 @TXTX,92,2,2,0,1,0,A$:T$=T$+A$:X=X+16:N=N+1:IFN=2T
HENX=X+16                                            -107-
80 IFN<4THEN50                                       -105-
90 TI$=T$+"00":@HRS6,1,0:@TXT88,80,2,2,16,1,0,"TEXT-D
EMO":T=9                                             -181-
100 TP$=TI$:@TXT144,100,2,2,16,1,0,"0"+RIGHT$(STR$(T)
,1):T=T-1                                            -169-
110 IFTP$=TI$THEN110                                 -200-
120 IFT>=0THEN100                                    -001-
130 T$="TEXT-DEMO":@HRS6,1,0:FORA=1TO9               -070-
140 A$=MID$(T$,A,1):@TXT144,20,4,4,0,1,0,A$:@TXT144,2
0,4,4,0,1,0," "                                      -245-
150 @TXT16+(A-1)*32,65,4,4,0,1,0,A$:NEXTA            -190-
160 @TXT0,150,2,3,16,1,0,"(C) GENARO NIETO '86":GOSUB
1000                                                 -161-
170 @DFC28,24,0,0,24,24,24,24,0                      -029-
180 @HRS6,1,0:@TXT24,72,2,2,16,1,0,"CON ESTA UTILIDAD
"                                                    -095-
190 @TXT16,92,2,2,16,1,0,"PUEDES ESCRIBIR EN"
                                                     -092-
200 @TXT24,112,2,2,16,1,0,"\ALTA RESOLUCION!":GOSUB10
00                                                   -205-
210 @HRS6,1,0:@TXT24,80,2,2,16,1,0,"TANTO DEL DERECHO
"                                                    -080-
220 @TXT272,120,2,2,16,3,0,"COMO DEL REVES":GOSUB1000
                                                     -011-
230 @HRS6,1,0:@TXT140,196,2,2,16,4,0,"HACIA ARRIBA"
                                                     -078-
240 @TXT180,12,2,2,16,2,0,"HACIA ABAJO":GOSUB1000
                                                     -229-
250 @HRS6,1,0:@TXT24,80,2,2,16,1,0,"EN VIDEO NORMAL O
"                                                    -065-
260 @TXT104,100,2,2,16,1,1,"INVERSO":GOSUB1000
                                                     -100-
270 @HRS6,1,0:@TXT88,45,2,2,16,1,0,"ELIGIENDO"
                                                     -092-
280 @TXT32,65,4,2,32,1,0,"ANCHURA,"                  -156-
290 @TXT96,85,2,3,16,1,0,"ALTURA Y"                  -190-
300 @TXT120,115,2,2,16,1,0,"COLOR"                   -064-
310 FORA=0TO15:@HRSA,1,1:FORB=0TO200:NEXTB,A:@HRS6,1,
1                                                    -056-
320 @TXT16,135,2,2,16,1,0,"DE LOS CARACTERES,":GOSUB1
000                                                  -212-
330 @HRS6,1,0:@TXT48,70,2,2,16,1,0,"DEFINIENDO LOS"
                                                     -166-
340 @TXT56,90,2,2,16,1,0,"TUYOS PROPIOS"             -095-
350 @TXT112,110,3,3,24,1,0,"@\!?":GOSUB1000:@HRS6,1,0
                                                     -144-
360 @TXT24,70,2,2,16,1,0,"Y ":@SET204:@TXT56,70,2,2,1
6,1,0,"PUDIENDO VARIAR"                              -012-
370 @TXT72,90,2,2,16,1,0,"EL JUEGO DE"               -068-
380 @TXT72,110,2,2,16,1,0,"CARACTERES.":@SET206
                                                     -163-
390 GOSUB1000:@HRS6,1,0:Y=20:FORA=1TO5:X=80:FORB=1TO5
                                                     -013-
400 @TXTX,Y,B,A,0,1,0,"A":X=X+10+8*B                 -244-
410 NEXTB:Y=Y+10+8*A:NEXTA:GOSUB1000                 -238-
420 SYS49152:@HRS6,1,0:@TXT32,60,2,2,16,1,0,"@ABCDEFG
HIJKLMNO"                                            -113-
```

```
430 @TXT32,80,2,2,16,1,0,"PQRSTUVWXYZ[\]^_"      -224-
440 @TXT32,100,2,2,16,1,0," !"+CHR$(34)+"#$%&'()*+,-.
/"                                                  -059-
450 @TXT32,120,2,2,16,1,0,"0123456789:;<=>?":GOSUB100
0:@HRS6,1,0                                         -222-
460 Y=20:FORA=1TO 4:@TXT(320-64*A)/2,Y,A,A,A*8,1,0,"T
U MICRO":Y=Y+8*A:NEXTA                              -135-
470 Y=180:FORA=1TO 4:@TXT320-(320-64*A)/2,Y,A,A,A*8,3
,0,"TU MICRO":Y=Y-8*A:NEXTA                         -133-
480 GOSUB1000:GOSUB1000:@HRS6,1,0:@TXT0,40,2,2,16,1,0
,"ESTA DEMOSTRACION HA"                             -254-
490 @TXT16,60,2,2,16,1,0,"TERMINADO. SON LAS":C=1
                                                    -058-
500 H$=LEFT$(TI$,2):M$=MID$(TI$,3,2):@TXT120,80,2,2,1
6,1,0,H$+" "+M$                                     -062-
510 IFS$=RIGHT$(TI$,2)THEN510                       -208-
520 S$=RIGHT$(TI$,2):C=ABS(C-1):C$=" ":IFC=0THENC$=":
"                                                   -011-
530 @TXT152,80,2,2,0,1,0,C$:IFM$<>MID$(TI$,3,2)THEN50
0                                                   -205-
540 GOTO510                                         -136-
1000 FORA=0TO999:NEXTA:RETURN                       -152-
READY.                                              -173-
```

PROGRAMA TEXT

```
10 F1=49152:F2=49252:T=12246:GOSUB60                -025-
20 F1=49280:F2=49297:T=1411:GOSUB60                 -237-
30 F1=49312:F2=49323:T=1859:GOSUB60                 -240-
40 F1=49408:F2=50248:T=102491:GOSUB60               -079-
50 SYS49152:NEW                                     -173-
60 S=0:FORA=F1TOF2:READDT:POKEA,DT:S=S+DT:NEXTA
                                                    -109-
70 IF S=TTHEN RETURN                                -073-
80 PRINT"ERROR EN DATAS":END                        -052-
97 REM **************                               -224-
98 REM * INTERPRETE *                               -043-
99 REM **************                               -226-
100 DATA 162,013,160,192,142,008,003,140,009,003,076,
016,196,032                                         -179-
101 DATA 115,000,240,004,201,064,240,003,076,231,167,
032,115,000                                         -156-
102 DATA 162,128,160,192,142,045,192,140,046,192,162,
000,160,000                                         -183-
103 DATA 177,122,217,131,192,208,035,200,192,003,208,
244,138,010                                         -189-
104 DATA 170,189,160,192,141,079,192,189,161,192,141,
080,192,032                                         -223-
105 DATA 115,000,032,115,000,032,115,000,032,095,193,
076,174,167                                         -178-
106 DATA 238,045,192,238,045,192,238,045,192,232,224,
006,208,198,076,008,175                             -250-
107 REM *****************                           -134-
108 REM * INSTRUCCIONES *                           -058-
109 REM *****************                           -136-
110 DATA 072,082,083,078,082,077,080,076,084,084,088,
084,083,069,084,068,070,067                         -206-
111 REM ************                                -175-
112 REM * PUNTEROS *                                -204-
113 REM ************                                -177-
114 DATA 000,193,095,193,117,193,184,193,149,193,156,
193                                                 -032-
115 REM ***********                                 -137-
116 REM * RUTINAS *                                 -118-
117 REM ***********                                 -139-
118 DATA 032,158,183,138,010,010,010,010,133,253
                                                    -096-
119 DATA 032,253,174,032,158,183,024,138,101,253,133,
253,032,253                                         -201-
120 DATA 174,032,158,183,138,208,020,160,000,162,160,
169,000,132                                         -185-
121 DATA 251,134,252,145,251,200,208,251,232,224,192,
208,244,165                                         -196-
122 DATA 253,153,000,128,153,200,128,153,144,129,153,
088,130,153                                         -195-
123 DATA 032,131,200,192,200,208,236,173,017,208,009,
032,141,017                                         -180-
124 DATA 208,173,000,221,041,253,141,000,221,169,008,
141,024,208                                         -175-
125 DATA 096,173,017,208,041,223,141,017,208,173,000,
221,009,002                                         -185-
126 DATA 141,000,221,169,021,141,024,208,096,032,138,
173,032,247                                         -192-
127 DATA 183,140,191,002,141,192,002,032,253,174,032,
158,183,142                                         -195-
128 DATA 193,002,032,253,174,032,158,183,134,010,076,
123,195,032                                         -202-
129 DATA 158,183,142,199,194,096,032,158,183,138,032,
178,194,160                                         -248-
130 DATA 000,132,255,032,253,174,032,158,183,164,255,
138,145,253.                                        -205-
131 DATA 200,192,008,208,238,096,032,138,173,032,247,
183,141,186                                         -214-
```

```
132 DATA 002,140,185,002,032,253,174,032,158,183,142,
187,002,032                                          -187-
133 DATA 253,174,032,158,183,142,178,002,032,253,174,
032,158,183                                          -213-
134 DATA 142,180,002,032,253,174,032,158,183,142,182,
002,032,253                                          -189-
135 DATA 174,032,158,183,138,041,001,141,183,002,138,
041,002,141                                          -185-
136 DATA 184,002,032,253,174,032,158,183,142,177,002,
032,253,174                                          -204-
137 DATA 032,158,173,036,013,048,003,076,008,175,032,
166,182,170                                          -210-
138 DATA 240,013,133,014,160,000,177,034,032,045,194,
198,014,208                                          -194-
139 DATA 001,096,230,034,208,002,230,035,076,024,194,
032,164,194                                          -199-
140 DATA 173,185,002,174,186,002,172,187,002,141,188,
002,142,189                                          -209-
141 DATA 002,140,190,002,174,180,002,142,181,002,164,
002,177,253                                          -175-
142 DATA 174,177,002,240,002,073,255,133,255,173,188,
002,174,189                                          -215-
143 DATA 002,172,190,002,141,191,002,142,192,002,140,
193,002,032                                          -167-
144 DATA 125,194,032,003,195,206,181,002,208,214,230,
002,164,002                                          -181-
145 DATA 192,008,208,200,076,056,195,169,008,072,024,
006,255,176                                          -222-
146 DATA 003,169,000,044,169,001,133,010,174,178,002,
142,179,002                                          -189-
147 DATA 032,123,195,032,206,194,206,179,002,208,245,
104,170,202                                          -200-
148 DATA 138,208,220,096,201,032,048,035,201,064,048,
006,201,096                                          -201-
149 DATA 016,027,041,191,162,000,134,002,134,254,010,
038,254,010                                          -178-
150 DATA 038,254,010,038,254,133,253,024,165,254,105,
206,133,254                                          -199-
151 DATA 096,076,008,175,173,183,002,240,035,173,184,
002,240,015                                          -203-
152 DATA 056,173,191,002,233,001,141,191,002,176,003,
206,192,002                                          -180-
153 DATA 096,024,173,191,002,105,001,141,191,002,144,
003,238,192                                          -187-
154 DATA 002,096,173,184,002,240,004,238,193,002,096,
206,193,002                                          -199-
155 DATA 096,173,183,002,208,035,173,184,002,240,015,
056,173,188                                          -216-
156 DATA 002,233,001,141,188,002,176,003,206,189,002,
096,024,173                                          -195-
157 DATA 188,002,105,001,141,188,002,144,003,238,189,
002,096,173                                          -205-
158 DATA 184,002,208,004,238,190,002,096,206,190,002,
096,173,183                                          -211-
159 DATA 002,240,037,173,184,002,240,016,056,173,185,
002,237,182                                          -203-
160 DATA 002,141,185,002,176,003,206,186,002,096,024,
173,185,002                                          -191-
161 DATA 109,182,002,141,185,002,144,003,238,186,002,
096,173,184                                          -204-
162 DATA 002,240,010,174,182,002,238,187,002,202,208,
250,096,174                                          -192-
163 DATA 182,002,206,187,002,202,208,250,096,173,192,
002,141,168                                          -201-
164 DATA 002,173,191,002,170,041,248,141,167,002,138,
041,007,141                                          -186-
165 DATA 169,002,173,193,002,170,041,007,141,170,002,
138,074,074                                          -196-
166 DATA 074,141,171,002,169,000,141,172,002,024,109,
171,002,109                                          -182-
167 DATA 171,002,109,171,002,109,171,002,109,171,002,
141,173,002                                          -177-
168 DATA 074,110,172,002,074,110,172,002,170,173,172,
002,024,109                                          -182-
169 DATA 167,002,168,138,109,168,002,024,105,160,170,
152,024,109                                          -207-
170 DATA 170,002,168,138,105,000,133,252,132,251,169,
001,141,174                                          -186-
171 DATA 002,056,169,007,237,169,002,170,014,174,002,
202,208,250                                          -193-
172 DATA 032,009,196,160,000,173,174,002,166,010,240,
007,017,251                                          -185-
173 DATA 145,251,076,009,196,073,255,049,251,145,251,
165,001,073                                          -221-
174 DATA 001,133,001,096,173,014,220,041,254,141,014,
220,165,001                                          -170-
175 DATA 041,251,133,001,160,000,162,208,169,206,132,
251,132,253                                          -184-
176 DATA 133,254,134,252,177,251,145,253,200,208,249,
230,254,232                                          -212-
177 DATA 224,210,208,240,165,001,009,004,133,001,173,
014,220,009                                          -175-
178 DATA 001,141,014,220,096                         -153-

READY.                                               -173-
```



TU MICRO COMMODORE ha creado el revolucionario sistema de introducción de programas FUERA ERRORES. Este nos permitirá introducir, sin temor alguno al esfuerzo inútil, cualquier listado por largo y complicado que parezca.

Para adoptar los listados publicados bajo este sistema, deberemos seguir las siguientes normas:

1) Es fundamental transcribir EXACTAMENTE el listado reproducido, incluyendo todos sus espacios, aunque se trate de separaciones entre número de instrucción y línea de instrucción.

2) Todas las líneas finalizarán con un número de tres dígitos, encerrado entre guiones, que NO deberá ser introducido, puesto que no forma parte del programa, sino que tiene la finalidad de hacer funcionar el sistema FUERA ERRORES, según veremos más adelante. Para evitar equivocaciones, dicha cifra entre guiones se sitúa en el margen derecho del final de la línea BASIC a la cual corresponde, a una distancia prudencial del mismo.

3) Para facilitar la introducción de símbolos difícilmente interpretables, se procede a la siguiente representación en los listados.

– Las letras aparecidas entre menor y mayor deberán ser introducidas con pulsación simultánea de la tecla COMMODORE y la letra representada. Ej.: < M > =COMMODORE M.

– Las letras aparecidas entre barras verticales deberán ser introducidas como pulsación simultánea de la tecla SHIFT y la letra representada. Ej.:|K|= SHIFT K.

– Entre corchetes simples se representarán los símbolos que se obtienen por pulsación directa de la tecla, aunque lógicamente, este caso sólo se dará para indicar las sucesiones de más de una letra. Así por ejemplo, la introducción de 5 asteriscos se representaría por [5*].

– Para la repetición de símbolos obtenidos mediante las teclas COMMODORE o SHIFT, se seguirá una combinación de las tres normas anteriormente citadas. Así por ejemplo, la introducción de 10 símbolos COMMODORE H, se representaría por [< 10 H >].

– Para evitar confusiones, cuando se utilice el sistema de representación de sucesiones de carácter, y éste sea un espacio, se utilizará la abreviatura ESP. [15 ESP] = 15 espacios.

– Los caracteres de control, tales como desplazamientos del cursor, colores, estados de reversa y funciones, se simbolizarán por una abreviatura de tres letras (dos más un espacio en el caso de las funciones) encerradas entre llaves, tal como se señala en la tabla adjunta.

Para introducir cualquier listado por el sistema FUERA ERRORES, deberemos entrar previamente y ejecutar el listado BASIC que aparece en la página siguiente por lo cual es recomendable conservar una copia grabada del mismo, para sucesivas ocasiones. Una vez introducido este listado, ya sea por el teclado, o a través de cinta o disco, debemos ejecutarlo con **RUN**. Instantes más tarde aparecerá en la pantalla el mensaje **FUERA ERRORES!** y el cursor libre para la entrada de programas, con el tradicional **READY**. Por encima, lo cual indicará la activación del sistema de depuración de errores.

En virtud al **NEW** que finaliza línea 20 del programa ¡FUERA ERRORES!, éste habrá desaparecido de la memoria, y seremos libres para introducir cualquiera de los programas listados en la sección **TECLA A TECLA** de cualquier número de nuestra revista, o incluso aquellos que apareciendo en otras secciones se acojan a este sistema. Así pues, si el programa ¡FUERA ERRORES! ha desaparecido de la memoria, ¿qué hemos conseguido ejecutándolo? Bien, la respuesta se llama informáticamente **INTERRUPCION**: se trata de una técnica de programación en código máquina que permite que el ordenador efectúe prácticamente dos trabajos a un tiempo, o más correctamente, que ejecute determinada tarea de forma automática, sin necesidad de que le prestemos una atención constante, de forma similar al proceso de respiración en un humano.

Efectivamente, aunque el soporte BASIC ha desaparecido de la memoria, antes de ‹marcharse› ha dejado funcionando en modo interrupción la pequeña rutina en código máquina que se hallaba en sus **DATAS**. Para comprobarlo pulsa **RETURN**; observarás algo muy extraño: tu ordenador no se comporta normalmente, no sólo desciende una línea el cursor, sino que además hace aparecer un número en la esquina superior izquierda de la pantalla. A continuación veremos como emplearlo.

Cada vez que pulsemos **RETURN**, aparecerá un número en la mencionada zona de la pantalla, y éste corresponderá con la instrucción que hayamos introducido. Esto forma parte del sistema de FUERA ERRORES. Cuando introduzcamos cualquier línea de un listado de este tipo, deberemos fijarnos en el número que aparece al pulsar **RETURN** de fin de línea; si éste coincide con el que aparece en el listado al final de la línea, ésta habrá sido introducida correctamente, en caso contrario existe algún error de tecleado que debemos modificar. Para modificar una instrucción errónea, no tenemos ni tan siquiera que volver a teclearla si no queremos, bastará sencillamente con modificar el carácter o caracteres erróneos como siempre hacemos, hasta que coincida el número de verificación que se presentará al pulsar **RETURN**.

Así pues, el sistema FUERA ERRORES se compone de dos partes: una codificación especial de los listados que facilitan su introducción, evitando los errores al confundir los caracteres gráficos, de control, etc., y un sistema de verificación de líneas que nos advierte en el preciso instante de introducir una de estas, que está mal tecleada.

Ahora bien, el empleo de estos dos sistemas no quiere decir que nos encontremos ante un BASIC diferente al de COMMODORE 64. Este no ha cambiado, hace exactamente las mismas cosas de siempre; simplemente hemos cambiado la forma de hacer los listados. En cuanto al misterioso número que aparece en la esquina de la pantalla no es más que una simple suma de comprobación, lo que se conoce técnicamente como un CHECKSUM. La rutina en código máquina de interrupción suma los valores de los caracteres que entra-

TABLA DE INTERPRETACION DE CODIGOS DE CONTROL					
ABR	***SIGNIFICADO***	***OBTENCION***	***ABR***	***SIGNIFICADO***	***OBTENCION***
HOM	HOME	CLR/HOME	WHT	WHITE (BLANCO)	CTRL 2
CLR	CLEAR + HOME	SHIFT CLR/HOME	RED	RED (ROJO)	CTRL 3
ABJ	CURSOR ABAJO	CRSR VERTICAL	CYN	CYAN (CIAN)	CTRL 4
ARB	CURSOR ARRIBA	SHIFT CRSR VERTICAL	PUR	PURPPLE (PURPURA)	CTRL 5
DCH	CURSOR DERECHA	CRSR HORIZONTAL	GRN	GREEN (VERDE)	CTRL 6
IZQ	CURSOR IZQUIERDA	SHIFT CSRS HORIZONTAL	BLU	BLUE (AZUL)	CTRL 7
RON	REVERSE ON	CTRL 9	YEL	YELLOW (AMARILLO)	CTRL 8
ROF	REVERSE OFF	CTRL 0	NRJ	NARANJA	COMMODORE 1
F1	FUNCION 1	F1	MRN	MARRON	COMMODORE 2
F2	FUNCION 2	SHIFT F1	RCL	ROJO CLARO	COMMODORE 3
F3	FUNCION 3	F3	GR1	GRIS 1	COMMODORE 4
F4	FUNCION 4	SHIFT F3	GR2	GRIS 2	COMMODORE 5
F5	FUNCION 5	F5	VCL	VERDE CLARO	COMMODORE 6
F6	FUNCION 6	SHIFT F5	ACL	AZUL CLARO	COMMODORE 7
F7	FUNCION 7	F7	GR3	GRIS 3	COMMODORE 8
F8	FUNCION 8	SHIFT F7	PI	π	SHIFT ↑
BLK	BLACK (NEGRO)	CTRL 1	^	↑	↑
			\	£	£
			STP	STOP	RUN/STOP
			INS	INSERT	INST/DEL
			—	←	←

mos en cada línea que introducimos, aplicándoles un módulo 256, es decir, volviendo a cero cada vez que su suma parcial supera el 255, de este modo se crea un número entre 0 y 255 dependiente directamente de los caracteres introducidos. Así pues, es prácticamente imposible que una sucesión de errores den por casualidad ese número, mientras que siempre que la línea esté correctamente introducida obtendremos el mismo código que nosotros hallamos por igual sistema y añadimos al final de cada línea al realizar el listado del programa.

Por tanto, el misterioso número no es tampoco una modificación del BASIC de COMMODORE, sino simplemente un pequeño truco para la comprobación de que las líneas han sido bien introducidas. Propiamente no nos evita cometer errores de tecleado, sino que simplemente nos advierte inmediatamente en qué línea los hemos introducido.

Para desactivar el sistema sólo deberemos pulsar **RUN/STOP+RESTORE**, y si por cualquier motivo nos interesara reactivarlo, podríamos ejecutar **SYS 822**, siempre y cuando se encuentre el código máquina en la memoria, lógicamente.

¡ADVERTENCIA! Puesto que el código máquina se encuentra ubicado en el buffer del casete, es imprescindible desactivarlo **(RUN/STOP+RESTORE)** antes de realizar cualquier operación con dicho periférico.

Utilización del casete con fuera errores

Si queremos introducir parte de un programa, para continuar posteriormente el trabajo emprendido, sin perder por supuesto la enorme ventaja del FUERA ERRORES deberemos proceder de la siguiente forma:

1. Desactivar el sistema FUERA ERRORES mediante la pulsación de RUN/STOP + RESTORE.
2. Trasladar el código máquina desde su ubicación en el buffer del casete hasta otro punto desocupado en la memoria RAM (por ejemplo 49152). Para ello sólo hemos de saber que ocupa 114 bytes desde 822 (inclusive). Una sencilla rutina que efectúe el trabajo de reubicación puede ser: **FORI=0TO113:POKE49152+I,PEEK(822+I):NEXT.**
3. Realizar la operación correspondiente con el casete; ya sea grabar una copia de seguridad de lo introducido en la memoria hasta el momento, ya sea cargar el casete una parte ya grabada del programa.
4. Restablecer FUERA ERRORES a su punto original. Siguiendo el caso de la línea de ejemplo anterior: **FORI=0TO113:POKE822+I,PEEK(49152+I):NEXT.**
5. Reactivar el sistema FUERA ERRORES mediante SYS 822 (esta vez no aparecerá mensaje alguno, aunque al pulsar **RETURN** comprobaremos que aparecen las cifras de control en la esquina superior izquierda). Lógicamente, los pasos 4 y 5 no son necesarios si después de una grabación se va a apagar el ordenador, y sólo se llevarán a cabo después de las cargas, o si tras la grabación de una copia de seguridad del programa, deseamos continuar introduciéndolo acto seguido.

```
10 FORI=822TO935:READA:C=C+A:POKEI,A:NEXT
20 IFC<>15254THENPRINTCHR$(147)"ATENCION!, HAY UN ERROR EN LOS DATOS":END
30 PRINTCHR$(147)TAB(213)"FUERA ERRORES!":SYS822:NEW
100 DATA 169,3,141,37,3,169,69,141,36,3,169,0,133,254,96,32,87,241,133,251
110 DATA 134,252,132,253,8,201,13,240,13,24,101,254,133,254,165,251,166,252
120 DATA 164,253,40,96,169,13,32,210,255,165,214,141,176,3,206,176,3,169,0
130 DATA 133,216,169,18,32,210,255,169,19,32,210,255,169,45,32,210,255,166
140 DATA 254,224,100,176,5,169,48,32,210,255,224,10,176,5,169,48,32,210,255
150 DATA 169,0,133,254,32,205,189,169,45,32,210,255,173,176,3,133,214,76,88,3
```


LOS 7 MAGNIFI
2. Skyfox
1. Commando
6. Harball



Colaboremos todos en la confección de estos 7 MAGNIFICOS de TU MICRO COMMODORE. Envíanos el nombre de tu programa favorito, dejando muy claro tu nombre y dirección. Todos los meses, sortearemos cinco suscripciones por un año a nuestra revista entre las respuestas recibidas. Anímate y escríbenos a: TU MICRO COMMODORE. (7 MAGNIFICOS). Apartado de Correos 61.294. 28080 MADRID.

3. Rambo

7. Mindshadow

5. Leader Board Golf

4. La ley del Oeste

The wall (el juego)

En esta sección de programación, vamos a ver un programa de juego típico de TECLA A TECLA. Aprovechando muchas de las técnicas divulgadas en TU MICRO COMMODORE como Chispas, Software, Tecla a tecla y Código máquina, hemos confeccionado The Wall, para divertirnos aprendiendo.

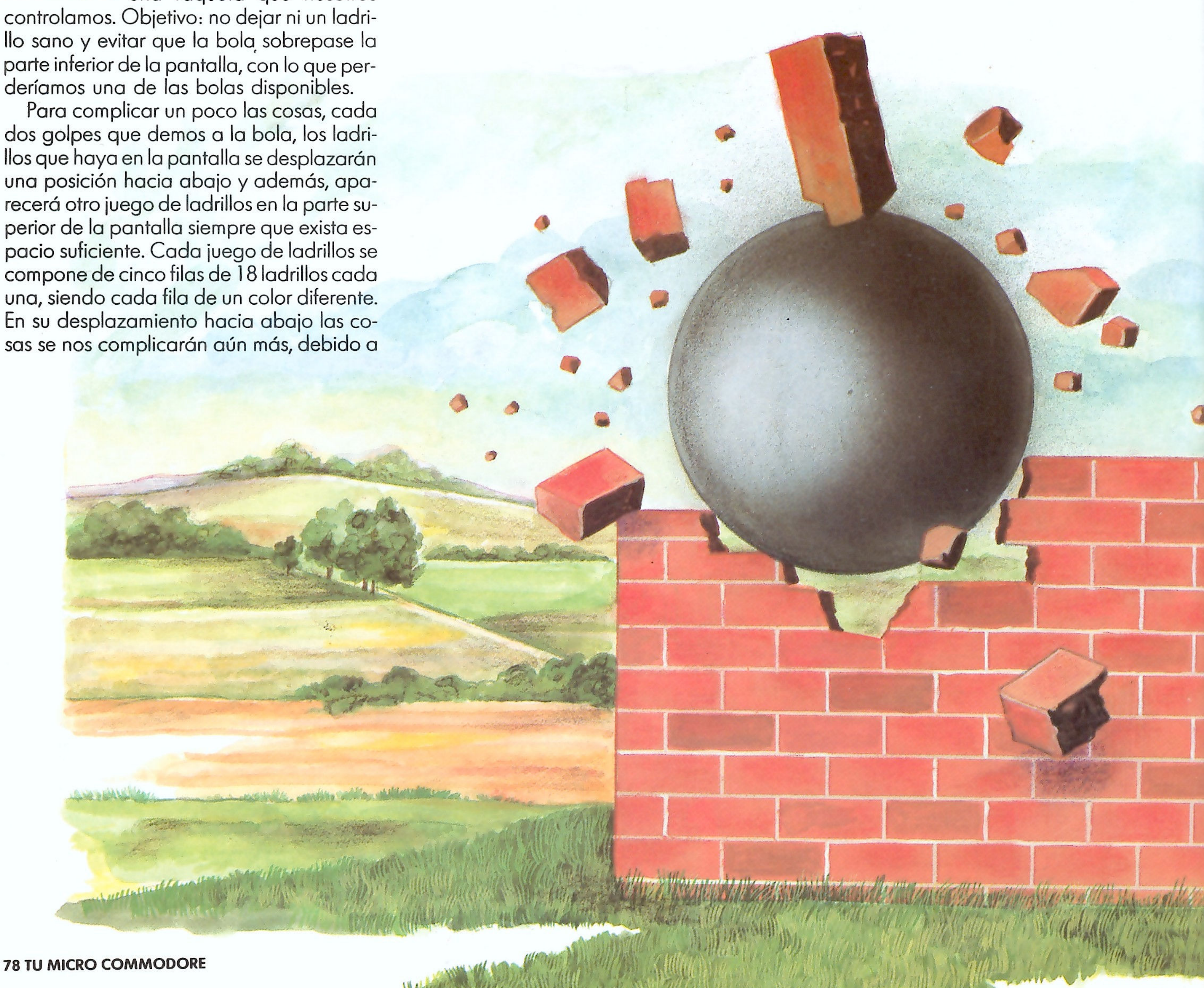
El programa es uno de los clásicos de los ordenadores domésticos; se trata de un muro de ladrillos, una bola y una raqueta que nosotros controlamos. Objetivo: no dejar ni un ladrillo sano y evitar que la bola sobrepase la parte inferior de la pantalla, con lo que perderíamos una de las bolas disponibles.

Para complicar un poco las cosas, cada dos golpes que demos a la bola, los ladrillos que haya en la pantalla se desplazarán una posición hacia abajo y además, aparecerá otro juego de ladrillos en la parte superior de la pantalla siempre que exista espacio suficiente. Cada juego de ladrillos se compone de cinco filas de 18 ladrillos cada una, siendo cada fila de un color diferente. En su desplazamiento hacia abajo las cosas se nos complicarán aún más, debido a la proximidad entre la pared y nuestra raqueta, llegando a caber en la pantalla tres juegos de ladrillos separados entre sí por una línea en blanco.

Pero vamos a explicar el programa detalladamente para ver los problemas que han surgido en su preparación y cómo se han ido resolviendo.

El muro de las dificultades

En la línea 100 ponemos el teclado en autorrepetición, para que al pulsar las teclas de control la raqueta se desplace uni-



formemente, y no por sucesivos golpes en la tecla (la primera chispa del primer número). También ajustamos los colores de fondo y borde.

Desde la línea 105 hasta la 175 se encuentran los PRINT que construyen la pantalla de presentación. Confeccionar una pantalla de este tipo es extremadamente complicado; observad la cantidad de espacios, reversos on y off, colores y caracteres gráficos empleados para ello. Gracias a uno de nuestros programas de TECLA A TECLA (PANTAMON número 17, primera época), el trabajo ha sido sencillísimo: simplemente cargar el programa, dibujar la pantalla y pulsar F5, para la autogeneración de las líneas PRINT correspondientes. Después se encadenan estas líneas al resto del programa (La Chistera número 2, primera época) y se numeran colocándolas en el sitio deseado.

La línea 180 se encarga de esperar hasta que se pulse una tecla, para continuar con la pregunta del número de bolas que deseamos por partida. Es importante observar la depuración que viene después de esta entrada, en 190 chequeamos que nb (número de bolas de juego) no sea menor que 1 ni mayor que 10 y en 195 que sea un número de bolas entero (no bola y media), ya que podría resultar fatal en la comprobación de la línea 460.

Las líneas 200-210 se encargan de mostrar en la pantalla los controles del juego; se aprovecha el tiempo de lectura para ubicar el código máquina en la memoria (215) y se añade un bucle de retardo al final para los que lean despacio.

Cada byte leído con READ, se suma a la variable ck, construyendo así una suma total de todo el código; si esa suma no coincide con la esperada (220-225) aparecerá un mensaje de error y se nos instará a usar correctamente el sistema FUERA ERRORES. Si se ha empleado este sistema es casi IMPOSIBLE tener un error en línea alguna, puesto que si el número aparecido en la pantalla no coincide con el del listado, debe existir algún error en la introducción, aunque no nos sea fácil localizarlo.

Seguimos con el dibujo en pantalla del borde del tablero (230-235) usando las líneas gráficas del teclado, pero escogiendo las que miran hacia dentro de la pantalla, para dar una sensación de rebote más real. Para colocar los ladrillos, hemos utilizado código máquina, debido a la rapidez que se necesitara para esta operación durante el juego (240). El código máquina se limita a colocar los ladrillos en la memoria de pantalla, con sus colores en la memoria de color correspondiente. Observa que los colores no están escogidos al azar, sino pensados para que en blanco y negro den una tonalidad decreciente (no todo el mundo dispone de una TV en color), baja el control de color y podrás comprobarlo; no es un esfuerzo considerable y muchos lo agradecen: ¡nunca elijas los colores que manejará un programa al azar!

En las líneas 245, 250 y 265 se definen en variables todas las constantes que se usarán durante el juego, ya que el C 64 opera más rápido con variables que con constantes que tiene que transformar cada vez que las encuentra (Software número 12, primera época). Las variables corresponden a las siguientes constantes: so=49342 dirección de salto para rutina de sonido, tr=17 forma de onda triangular, ru=33 forma de onda diente de sierra, mu=49335 dirección de cambio para forma de onda, a$ es la raqueta formada por tres caracteres diferentes para detectar la dirección de rebote, b$ borrará la raqueta cuando sea necesario, x=1 posición inicial de la raqueta, k=54272 constante de color para sumar a la posición de pantalla, a=41 primera dirección de rebote (baja hacia la derecha, b=−41 sube hacia la izquierda, c=−39 sube hacia la derecha, d=39 baja hacia la izquierda, n=1024 primera posición de pantalla, v=40 control de línea inferior a la bola, s=99 código de carácter central de la raqueta.



La línea 255 borra la raqueta y espera a que se pulse la barra espaciadora, la línea 260 coloca la raqueta en posición de saque así como la bola en 1946, la dirección de subida se determina aleatoriamente. En 265 seguimos definiendo variables: e=32 es el valor del carácter blanco, t=100 es el carácter de la línea superior, m=103 pared izquierda, g=101 pared derecha, o=207 carácter izquierdo del ladrillo, l=208 carácter derecho del ladrillo, f=1984 posición de pérdida de bola, h=36 posición extrema de la raqueta por la derecha, w=81 carácter de la bola, u=120 carácter derecho de la raqueta, r=119 carácter izquierdo de la raqueta.

La línea 270 recoge la pulsación de una tecla en k$. En 275 hay una línea interesante, si borras el REM del principio y dejas la línea así: 275 ifythen..., la raqueta seguirá automáticamente a la bola y podrás presumir de una habilidad excepcional, para tus amigos coloca de nuevo el REM. La técnica empleada consiste en sumar a x (posición de la raqueta) uno o menos uno según la dirección de la bola, usando el valor de las comparaciones (−1 si es verdadero y 0 si es falso). En 280-285 se comprueba si se ha de mover la raqueta y se actualiza la posición teniendo como tope 1 y h (290-295); se «escribe» la raqueta en 300 para que siempre esté en pantalla.

En 305 tenemos una variable «interruptor», observad como cada vez que pase el programa por aquí, y valdrá 1 si valía 0 ó 0 si valía 1, de esta manera podemos mover la raqueta dos veces más rápido que la bola para poder llegar aunque estemos descolocados, las líneas 310-315 se encargan de saltar cuando y vale 1.

320 se encarga de borrar la bola, aumentar la posición, ver lo que hay en esa nueva posición (z), colocar la bola, y su color. Se comprueba si la bola ha salido por arriba o por abajo (la bola se puede escapar por las esquinas superiores), y si es así va a 460, que comprueba si quedan más bolas para después esperar la pulsación de espacio para sacar.

Si en las posiciones importantes de la bola hay un espacio (32), es inútil comprobar nada más, así que se salta todas las comprobaciones posteriores dando la agilidad al programa, es una línea importantísima (335). También en 340 vemos si podemos saltarnos un importante grupo de comprobaciones; de no ser así, desde 345 hasta 380 se comprueba la colisión de la bola con un ladrillo, la dirección de llegada, la dirección de salida, se actualiza la puntuación y se genera el sonido de choque con ladrillo (pokeso, ru:sysmu), una vez hecho esto se salta todo lo demás para agilizar los trámites. Las líneas 385-395 se encargan de hacer lo propio con la raqueta y las líneas 420-445 con las paredes del recinto.

En 405 se comprueba si es un golpe par, para hacer el «scroll» de la pantalla hacia abajo y en 410 si hay sitio para colocar otro juego de ladrillos (cada 12 golpes) sumándose 90 puntos de bonificación. En 415 se salta otro grupo de IF si no es necesario, la línea 450 coloca la bola y su color nuevamente y en 455 se vuelve al principio para repetir el bucle; a partir de aquí tenemos la subrutina de comprobación de última bola (460-465) y el salto a impresión de puntos y petición de otra partida (470-500).

La última parte del programa, tiene las líneas DATA necesarias para generar el código máquina utilizado en la colocación de un juego de ladrillos, el scroll hacia abajo de la pantalla y el sonido de los rebotes en las paredes, raqueta y ladrillos.

```
100 POKE650,128:POKE53280,6:POKE53281,0            -215-
105 PRINT"{CLR}{ABJ}[8 ESP]{YEL}[3 ESP]<D><4 I><F><D>
<F>[2 ESP]<D><F><D><4 I><F>"                       -021-
110 PRINT"[11 ESP]{RON}{NRJ}[5 ESP]{ROF}{YEL}<V>{RON}
{NRJ} {ROF}{YEL}<K>[2 ESP]{RON}{NRJ} {ROF}{YEL}<K>{RO
N}{NRJ}[5 ESP]{ROF}{YEL}<V>"                       -109-
115 PRINT"[13 ESP]{RON}{NRJ} {ROF}{YEL}<K>[2 ESP]{RON
}{NRJ} {YEL}<C>{ROF}<2 I>{RON}{NRJ} {ROF}{YEL}<K>{RON
}{NRJ} {YEL}<C>{ROF}<2 I><F>"                      -241-
120 PRINT"[13 ESP]{RON}{NRJ} {ROF}{YEL}<K>[2 ESP]{RON
}{NRJ}[5 ESP]{ROF}{YEL}<K>{RON}{NRJ}[4 ESP]{ROF}{YEL}
<V>"
                                                   -237-
125 PRINT"[13 ESP]{RON}{NRJ} {ROF}{YEL}<K>[2 ESP]{RON
}{NRJ} {ROF}{YEL}<K>[2 ESP]{RON}{NRJ} {ROF}{YEL}<K>{R
ON}{NRJ} {YEL}<C>{ROF}<3 I><F>"                    -117-
130 PRINT"[13 ESP]{RON}{NRJ} {ROF}{YEL}<V>[2 ESP]{RON
}{NRJ} {ROF}{YEL}<V>[2 ESP]{RON}{NRJ} {ROF}{YEL}<V>{R
ON}{NRJ}[5 ESP]{ROF}{YEL}<V>{2 ABJ}"               -203-
135 PRINT"[9 ESP]{CYN}<D><F>[2 ESP]<D><F><D><4 I><F><
D><F>[4 ESP]{CYN}<D><F>"                           -051-
140 PRINT"[9 ESP]{RON}{BLU} {ROF}{CYN}<K>[2 ESP]{RON}
{BLU} {ROF}{CYN}<K>{RON}{BLU}[5 ESP]{ROF}{CYN}<K>{RON
}{BLU} {ROF}{CYN}<K>[4 ESP]{RON}{BLU} {ROF}{CYN}<K>"
                                                   -149-
145 PRINT"[9 ESP]{RON}{BLU} {ROF}{CYN}<K>[2 ESP]{RON}
{BLU} {ROF}{CYN}<K>{RON}{BLU} {CYN}<C>{ROF}<2 I>{RON}
{BLU} {ROF}{CYN}<K>{RON}{BLU} {ROF}{CYN}<K>[4 ESP]{RO
N}{BLU} {ROF}{CYN}<K>"                             -156-
150 PRINT"[9 ESP]{RON}{BLU} {ROF}{CYN}<K><D><F>{RON}{
BLU} {ROF}{CYN}<K>{RON}{BLU}[5 ESP]{ROF}{CYN}<K>{RON}
{BLU} {ROF}{CYN}<K>[4 ESP]{RON}{BLU} {ROF}{CYN}<K>"
                                                   -061-
155 PRINT"[9 ESP]{RON}{BLU} {CYN}<C>{BLU} {CYN}<C>{BL
U} {ROF}{CYN}<V>{RON}{BLU} {ROF}{CYN}<K>[2 ESP]{RON}{
BLU} {ROF}{CYN}<K>{RON}{BLU} {CYN}<C>{ROF}<3 I><F>{RO
N}{BLU} {CYN}<C>{ROF}<3 I><F>"                     -228-
160 PRINT"[9 ESP]{BLU}<C>{RON} {ROF}<U>{RON}{BLU} {RO
F}<V> {RON}{BLU} {ROF}{CYN}<V>[2 ESP]{RON}{BLU} {ROF}
{CYN}<V>{RON}{BLU}[5 ESP]{ROF}{CYN}<V>{RON}{BLU}[5 ES
P]{ROF}{CYN}<V>{YEL}"                              -158-
165 PRINT"{3 ABJ}[9 ESP]{GRN}POR[3 ESP]ANTONIO[2 ESP]
CARVAJAL{ABJ}"                                     -252-
170 PRINT"[9 ESP]{RED}PARA TU MICRO COMMODORE{GR2}{AB
J}"                                                -186-
175 PRINT"[12 ESP]{RON} PULSA UNA TECLA {ROF}"
                                                   -004-
180 GETP$:IFP$=""THEN180                           -147-
185 PRINT"{CLR}{GR2}"SPC(200)"CUANTAS BOLAS POR PARTI
DA (1-10)";:INPUTNB                                -098-
190 IFNB<1ORNB>10THEN100                           -214-
195 IFNB<>INT(NB)THEN185                           -241-
200 PRINTSPC(205)" PULSA {RON} ESPACIO {ROF} PARA SAC
AR"                                                -204-
205 PRINTSPC(15)"{RON} > {ROF}(.) PARA DERECHA"
                                                   -022-
210 PRINTSPC(15)"{RON} < {ROF}(,) PARA IZQUIERDA"
                                                   -208-
215 FORI=49152TO49377:READA:CK=CK+A:POKEI,A:NEXT:FORK
=1TO2000:NEXT                                      -193-
220 IFCK<>31888THENPRINT"{CLR}HAY UN ERROR EN LINEAS
DATA. REPASALO"                                    -167-
225 IFCK<>31888THENPRINT"USA {RON} !FUERA ERRORES! {R
OF}":END                                           -081-
230 PRINT"{HOM} ";:FORI=1TO38:PRINT"<@>";:NEXT:PRINT"
 ";                                                -163-
235 PRINT"{ROF}";:FORI=1TO23:PRINT"<M>[38 ESP]<G>";:N
EXT                                                -003-
240 SYS49204                                       -184-
245 SO=49342:TR=17:RU=33:MU=49335                  -202-
250 A$="<Y><T><U>{ARB}":B$="[3 ESP]{ARB}":X=1:K=54272
:A=41:B=-41:C=-39:D=39:N=1024:V=40:S=99            -243-
255 PRINTTAB(X)B$:GETK$:IFK$<>" "THEN255           -153-
260 X=1:PRINTTAB(X)A$:P=1946:Q=-41:IFRND(.)>.5THENQ=-
39                                                 -066-
265 E=32:T=100:M=103:G=101:O=207:L=208:F=1984:H=36:W=
81:U=120:R=119                                     -025-
270 GETK$                                          -008-
275 REMIFYTHENPRINTTAB(X)B$:X=X+(Q=-41)-(Q=41)+(Q=39)
-(Q=-39)                                           -123-
280 IFK$=","THENPRINTTAB(X)B$:X=X-1                -140-
285 IFK$="."THENPRINTTAB(X)B$:X=X+1                -145-
290 IFX<1THENX=1:PRINTTAB(X)A$                     -176-
295 IFX>HTHENX=H:PRINTTAB(X)A$                     -229-
300 PRINTTAB(X)A$                                  -037-
305 Y=-(Y<>1)                                      -208-
310 IFYTHEN320                                     -096-
315 GOTO270                                        -139-
```

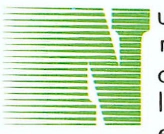
```
320 POKEP,E:P=P+Q:Z=PEEK(P):POKEP,W:POKEP+K,1
                                                   -151-
325 IFP>=FTHENPOKEP,E:GOSUB460:GOTO255             -220-
330 IFP<=NTHENPOKEP,E:GOSUB460:GOTO255             -222-
335 IFPEEK(P+Q)=32ANDPEEK(P+V)=32ANDZ=32THEN270
                                                   -129-
340 IFZ=32THEN385                                  -017-
345 IFZ=OANDQ=CTHENPOKEP,E:POKEP+1,E:Q=A:PU=PU+1:POKE
SO,RU:SYSMU:GOTO270                                -101-
350 IFZ=OANDQ=BTHENPOKEP,E:POKEP+1,E:Q=D:PU=PU+1:POKE
SO,RU:SYSMU:GOTO270                                -099-
355 IFZ=OANDQ=DTHENPOKEP,E:POKEP+1,E:Q=B:PU=PU+1:POKE
SO,RU:SYSMU:GOTO270                                -104-
360 IFZ=OANDQ=ATHENPOKEP,E:POKEP+1,E:Q=C:PU=PU+1:POKE
SO,RU:SYSMU:GOTO270                                -098-
365 IFZ=LANDQ=CTHENPOKEP,E:POKEP-1,E:Q=A:PU=PU+1:POKE
SO,RU:SYSMU:GOTO270                                -102-
370 IFZ=LANDQ=BTHENPOKEP,E:POKEP-1,E:Q=D:PU=PU+1:POKE
SO,RU:SYSMU:GOTO270                                -100-
375 IFZ=LANDQ=DTHENPOKEP,E:POKEP-1,E:Q=B:PU=PU+1:POKE
SO,RU:SYSMU:GOTO270                                -105-
380 IFZ=LANDQ=ATHENPOKEP,E:POKEP-1,E:Q=C:PU=PU+1:POKE
SO,RU:SYSMU:GOTO270                                -099-
385 IFPEEK(P+V)=RORPEEK(P+Q)=RTHENQ=B:GP=GP+1:POKESO,
TR:SYSMU:GOTO405                                   -091-
390 IFPEEK(P+V)=UORPEEK(P+Q)=UTHENQ=C:GP=GP+1:POKESO,
TR:SYSMU:GOTO405                                   -094-
395 IFPEEK(P+V)=SORPEEK(P+Q)=STHENGP=GP+1:POKESO,TR:S
YSMU:GOTO405                                       -084-
400 GOTO415                                        -135-
405 IFGP/2=INT(GP/2)THENSYS49152                   -228-
410 IFGP/12=INT(GP/12)THENSYS49204:PU=PU+90        -149-
415 IFPEEK(P+Q)=32THEN270                          -245-
420 IFPEEK(P+Q)=MANDQ=BTHENQ=C:POKESO,TR:SYSMU:GOTO27
0                                                  -120-
425 IFPEEK(P+Q)=MANDQ=DTHENQ=A:POKESO,TR:SYSMU:GOTO27
0                                                  -125-
430 IFPEEK(P+Q)=GANDQ=CTHENQ=B:POKESO,TR:SYSMU:GOTO27
0                                                  -115-
435 IFPEEK(P+Q)=GANDQ=ATHENQ=D:POKESO,TR:SYSMU:GOTO27
0                                                  -120-
440 IFPEEK(P+Q)=T ANDQ=CTHENQ=A:POKESO,TR:SYSMU:GOTO2
70                                                 -160-
445 IFPEEK(P+Q)=T ANDQ=BTHENQ=D:POKESO,TR:SYSMU:GOTO2
70                                                 -167-
450 POKEP,W:POKEP+K,1                              -071-
455 GOTO270                                        -144-
460 BO=BO+1:IFBO=NBTHENGOTO470                     -159-
465 RETURN                                         -159-
470 PRINT"{HOM}"SPC(200)SPC(215)"{RON}{GR2} SE ACABO
":P$=MID$(STR$(PU),2)                              -030-
475 PRINTTAB(10)"{RON} HAS HECHO "P$" PUNTOS "
                                                   -176-

480 PRINTTAB(10)"{RON} OTRA PARTIDA? (S/N) "
                                                   -067-
485 GETP$:IFP$=""THEN485                           -163-
490 IFP$="S"THENRUN                                -184-
495 IFP$="N"THENPRINT"{CLR}":END                   -056-
500 GOTO485                                        -143-
505 DATA 160,240,185,221,5,153,5,6,185,221,217,153,5,
218,136,208                                        -139-
510 DATA 241,160,240,185,237,4,153,21,5,185,237,216,1
53,21,217,136                                      -234-
515 DATA 208,241,160,200,185,39,4,153,79,4,185,39,216
,153,79,216                                        -163-
520 DATA 136,208,241,96,160,36,169,208,153,81,4,169,2
,153,81,216                                        -156-
525 DATA 136,169,207,153,81,4,169,2,153,81,216,136,20
8,232,160,36                                       -203-
530 DATA 169,208,153,121,4,169,4,153,121,216,136,169,
207,153,121,4                                      -239-
535 DATA 169,4,153,121,216,136,208,232,160,36,169,208
,153,161,4,169                                     -047-
540 DATA 5,153,161,216,136,169,207,153,161,4,169,5,15
3,161,216,136                                      -247-
545 DATA 208,232,160,36,169,208,153,201,4,169,3,153,2
01,216,136,169                                     -041-
550 DATA 207,153,201,4,169,3,153,201,216,136,208,232,
160,36,169,208                                     -030-
555 DATA 153,241,4,169,7,153,241,216,136,169,207,153,
241,4,169,7                                        -159-
560 DATA 153,241,216,136,208,232,96,162,15,138,141,24
,212,169,33,141                                    -084-
565 DATA 4,212,169,15,141,5,212,169,40,141,1,212,169,
200,141,0                                          -021-
570 DATA 212,160,255,136,208,253,202,208,224,169,0,14
1,4,212,141,5,212,96                               -060-
```



Slalom

Si nuestros esquiadores no logran cosechar ninguna medalla en la próxima Olimpiada de invierno, seguramente se deba a la escasez de pistas de entrenamiento de que adolece nuestro país. Pues bien..., ¡se acabaron los problemas!... excepto los derivados de la necesidad de aprovisionarse de vitrinas suficientes en las que exhibir los abundantes trofeos que se avecinan. El programa que presentamos traslada la blanca pista nevada a la pantalla de un televisor doméstico... ¿el equipo?..., cómodo y económico: un mullido sillón, un Commodore 64 ó 128 y toda la imaginación de la que seamos capaces.



Nuestro juego consiste, ni más ni menos, que en una simulación de esta elegante modalidad del deporte blanco. El esquiador se desliza por una pendiente a toda velocidad, sorteando —¡al menos de eso se trata!— unos obstáculos oportunamente distribuidos a lo largo del recorrido

Todos sabemos que en la competición deportiva tales obstáculos no son más que banderitas que encajan con resignación los baquetazos del esforzado esquiador. Obstáculo muy adecuado si lo que se pretende es ofrecer un cómodo trazado para la práctica de los más experimentados. En una de las pruebas propuestas por nosotros pretendemos avanzar un poco más, convirtiendo a nuestra pista en un trazado idóneo para el aprendizaje autodidacta, incluso del más neófito. Sin ir más lejos, en aras del interés didáctico y en beneficio de los profesionales del vendaje y la escayola, hacemos obsequio de las banderitas a las «mayoretes» y en su lugar plantamos una recia variedad de pinos nórdicos... Así que ¡no apuremos los giros so pena de acariciar a las susodichas banderitas!

Funcionamiento del programa

Garantizando el riesgo y la emoción del juego, pasemos a examinar el desarrollo del programa. Al iniciar su ejecución, un menú principal nos da la bienvenida, solicitándonos que indiquemos el número de participantes en las pruebas (1 por defecto) y el grado de experiencia de los mismos: principiantes o expertos.

Una vez cumplimentado este último trámite, podemos pasar a las pruebas en cuestión. En la primera de ellas, nuestras evoluciones por la pista serán controladas por el joystick, siguiendo unas sencillas reglas: los desplazamientos izquierdo y derecho de la palanca producen los giros correspondientes del esquiador, y empujar hacia adelante el joystick frena su marcha, al tiempo que el botón de disparo la acelera. La segunda prueba, tiene un manejo muy similar, también a través del joystick, con la única diferencia que para el frenado no es preciso el desplazamiento de la palanca, sino que es suficiente con dejar de acelerar (botón de disparo).

Por último, la tecla de función F7 servirá para retornar al menú principal durante la ejecución del programa. ¡Suerte! y cuidado con las banderitas...

LISTADO

```
30 :                                                    -189-
40 :                                                    -190-
45 DIMPS(25)                                            -190-
50 POKE53280,1:POKE53281,1:POKE53269,0:P$(1)="PLAYER"
:POKE650,150:RESTORE                                    -076-
55 E$="{4 ABJ}":P$(2)=P$(1):POKE53264,0                 -229-
56 M$(1)="JOYSTICK (PORT 2)":M$(2)="TECLADO[10 ESP]":
M=2:P=1:L=2                                             -069-
57 FORQ1=1TO7:READCO$(Q1):NEXT                          -149-
59 L$(1)="NOVICE":L$(2)="EXPERT"                        -075-
60 PRINT"{CLR}{BLK}"TAB(8)"{RON}<A>¦3 *¦{2 DCH}<A>¦2
*¦<S>{2 DCH}<A>¦2 *¦<S>{2 DCH}¦-¦{2 DCH}¦-¦{2 DCH}¦-¦
{ROF}"                                                  -039-
70 PRINTTAB(8)"{RON}¦-¦{5 DCH}¦-¦{5 DCH}¦-¦{2 DCH}¦-¦
{2 DCH}¦-¦{2 DCH}¦-¦{2 DCH}¦-¦{ROF}"                    -113-
80 PRINTTAB(8)"{RON}<Q>¦3 *¦{2 DCH}<Z>¦2 *¦<S>{2 DCH}
¦-¦{2 DCH}¦-¦{2 DCH}¦-¦{2 DCH}¦-¦{2 DCH}¦-¦{ROF}"
                                                        -208-
90 PRINTTAB(8)"{RON}¦-¦{8 DCH}¦-¦{2 DCH}¦-¦{DCH}<A><W
>{2 DCH}¦-¦{2 DCH}¦-¦{2 DCH}¦-¦{ROF}"                   -220-
100 PRINTTAB(8)"{RON}<Z>¦3 *¦{2 DCH}¦3 *¦<X>{2 DCH}<Z
>¦2 *¦<X>{ABJ}{IZQ}<Z>¦*¦{ARB}{DCH}<Z>¦2 *¦<X>{2 DCH}
¦-¦{ROF}"                                               -152-
110 PRINT,"{2 ABJ}{2 DCH}BY[2 ESP]* JORDI SAIS * "
                                                        -098-
120 PRINT"{2 ABJ}{BLU}","{RON}[7 ESP]{ABJ}{7 IZQ}[2 E
SP]F 1[2 ESP]{ABJ}{7 IZQ}[7 ESP]"                       -214-
130 PRINT"{2 ABJ}","{RON}[7 ESP]{ABJ}{7 IZQ}[2 ESP]F
3[2 ESP]{ABJ}{7 IZQ}[7 ESP]"                            -186-
140 PRINT"{2 ABJ}","{RON}[7 ESP]{ABJ}{7 IZQ}[2 ESP]F
5[2 ESP]{ABJ}{7 IZQ}[7 ESP]"                            -189-
145 PRINT"{HOM}{ABJ}"                                   -175-
150 FORQ=1TO9:PRINT:NEXTQ:PRINT,,CO$(CO)"{IZQ}"P;P$(2
):PRINTE$,,L$(L)                                        -121-
155 PRINTE$,,"START"                                    -219-
160 O$="":GETO$:IFO$="{F1 }"THENP=P+1:IFP=9THENP=1
                                                        -094-
170 IFO$="{F3 }"THENL=L+1:IFL=3THENL=1                  -143-
180 IFO$="{F5 }"THENPRINT"{CLR}":GOTO200                -091-
181 CO=CO+1:IFCO=8THENCO=1                              -118-
184 P$(2)=P$(1)+"S"                                     -169-
185 IFP=1THENP$(2)=P$(1)+"[2 ESP]"                      -019-
190 GOTO145                                             -141-
193 :                                                   -247-
194 :                                                   -248-
200 B$(1)="{RON}{NRJ}[2 ESP]{ARB}{4 IZQ}{GRN}¦\¦ {NRJ
}[2 ESP]{GRN} <*>{ARB}{6 IZQ}¦\¦[4 ESP]<*>{ARB}{5 IZQ
}¦\¦[2 ESP]<*>{ARB}{4 IZQ}¦\¦[2 ESP]<*>{ARB}{3 IZQ}¦\
¦<*>{ROF}"                                              -184-
201 B$(7)="{BLU}{RON}[13 ESP]{13 IZQ}"                  -236-
202 B$(8)="{ABJ} S A L I D A {ABJ}{13 IZQ}[13 ESP]{AB
J}{13 IZQ}"                                             -119-
203 B$(9)="{ROF}<13 Y><H>{IZQ}{ABJ}<H>{IZQ}{ABJ}<H>{A
BJ}{IZQ}<H>{ABJ}{IZQ}<H>{4 ARB}{15 IZQ}<N>{ABJ}{IZQ}<
N>{ABJ}{IZQ}<N>{ABJ}{IZQ}<N>{ABJ}{IZQ}<N>{ROF}"
                                                        -082-
204 B$(6)=B$(7)+B$(8)+B$(9)                             -003-
205 B$(7)="<13 P>{ABJ}{14 IZQ}<N><13 +><H>{15 IZQ}"
                                                        -175-
206 B$(10)="{ABJ}<N><3 +>M<+>E<+>T<+>A<3 +><H>{ABJ}{1
5 IZQ}<N><13 +><H>{ABJ}{14 IZQ}"                        -116-
207 B$(10)=B$(7)+B$(10)+B$(9):PR=1                      -164-
208 IFL=1THENLI=20:LO=26:DY=4                           -040-
210 IFL=2THENLI=25:LO=29:DY=3                           -041-
211 B$(2)="{RON}{ACL}¦3 ACL¦{ARB}{4 IZQ}¦5 IZQ¦{ARB}{
6 IZQ}¦7 IZQ¦{ARB}{6 IZQ}¦5 IZQ¦{ARB}{4 IZQ}¦3 IZQ¦{R
OF}"                                                    -168-
220 B$(4)=B$(1):B$(5)=B$(1)                             -015-
230 B$(3)="{RON}{NRJ} ¦2 -¦[3 ESP]{ARB}{6 IZQ} <A><S>
 <T> {ARB}{6 IZQ}[3 ESP]<N>{BLK}¦+¦{NRJ}<H>{ARB}{7 IZ
Q}¦\¦[4 ESP]<@> {ARB}{6 IZQ}{ROF}{BLK}<6 \>"
                                                        -144-
240 H$="{HOM}{25 ABJ}"                                  -095-
242 PRINT"{CLR}{11 ABJ}"TAB(15)"{RED}GET READY"
                                                        -058-
244 :                                                   -244-
245 :                                                   -245-
260 POKE2040,192:POKE2041,193:POKE2042,194:POKE2043,1
95:POKE2044,196                                         -101-
270 V=53248:POKE2045,197:POKE2046,198:POKE2047,199:KL
=350                                                    -001-
280 FORQ1=12288TO12350:READA:POKEQ1,A:NEXT              -107-
285 FORQ1=12352TO12414:READA:POKEQ1,A:NEXT              -105-
290 FORQ1=12416TO12478:READA:POKEQ1,A:NEXT              -112-
295 FORQ1=12480TO12542:READA:POKEQ1,A:NEXT              -110-
300 FORQ1=12544TO12606:READA:POKEQ1,A:NEXT              -099-
305 FORQ1=12608TO12670:READA:POKEQ1,A:NEXT              -106-
310 FORQ1=12672TO12734:READA:POKEQ1,A:NEXT              -104-
320 FORQ1=12736TO12798:READA:POKEQ1,A:NEXT              -116-
325 :                                                   -244-
326 :                                                   -245-
340 POKEV+28,255:POKEV+37,3:POKEV+38,6                  -003-
350 FORVV=39TO46:POKEV+VV,10:NEXTVV                     -073-
385 X=165:Y1=107:Y2=128:SP=1:PX=1:X2=2                  -245-
390 :                                                   -246-
395 :                                                   -251-
410 FORVV=1TO13STEP4:POKEV+VV,Y1:NEXTVV:FORVV=3TO15ST
EP4:POKEVV+V,Y2:NEXTVV                                  -198-
420 POKEV,X:POKEV+2,X                                   -095-
970 :                                                   -250-
980 :                                                   -251-
992 X2(1)=2:X2(2)=8:X2(3)=32:GG=0                       -026-
993 XV(1)=3:XV(2)=12:XV(3)=48                           -133-
995 FORQ2=1TOP:FF=0                                     -197-
1000 PRINT"{CLR}":CL=0:CT=0:X=165:CF=0:POKEV+38,6
                                                        -011-
1002 POKEV+21,3                                         -085-
1003 PRINT,"{3 DCH}"B$(6):VE=200:DX=0                   -012-
1005 TI$="[6 0]"                                        -072-
1006 PO=5:FORVC=1TO350                                  -091-
1007 GETTE$:IFTE$="{F7 }"THENRUN                        -056-
1008 ONSPGOTO1010,1040,1070                             -009-
1010 POKEV,X:POKEV+2,X:POKEV+21,3+FF:GOTO1090
                                                        -043-
1040 POKEV+4,X:POKEV+6,X:POKEV+21,12+FF:GOTO1090
                                                        -193-
1070 POKEV+8,X:POKEV+10,X:POKEV+21,48+FF                -191-
1090 IFPR=2THENRETURN                                   -153-
1095 IFINT(VC/DY)<>VC/DYTHEN1200                        -240-
1110 AZ=INT(RND(1)*6)+1:IFAZ=6THENAZ=5                  -044-
1120 PZ=INT(RND(1)*30)+1                                -086-
1125 IFPZ>LOTHEN1200                                    -237-
1131 IFPZ<LITHEN1135                                    -233-
1132 OP=INT(RND(1)*2)+1                                 -029-
1133 IFOP=1THENAP=4:GOTO1180                            -242-
1134 IFOP=2THENAP=34:GOTO1180                           -039-
1135 DI=ABS(AP-(PZ+PO))                                 -094-
1140 IFDI>9THEN1175                                     -118-
1150 IFDI<5THENCA=9:GOTO1160                            -216-
1155 CA=5                                               -226-
1160 IFPZ+CA+PO>28THEN1165                              -062-
1161 AP=PZ+PO+CA:GOTO1180                               -023-
1165 AP=(PZ+PO)-(2*CA):GOTO1180                         -027-
1175 AP=PO+PZ                                           -048-
1180 PRINTTAB(AP)B$(AZ)                                 -130-
1190 :                                                  -037-
1195 :                                                  -042-
1200 JV=PEEK(56320):FR=JVAND16                          -031-
1210 JV=15-(JVAND15)                                    -126-
1215 IFJV=1THENVE=VE+50                                 -184-
1220 IFFR<>16THENVE=VE-15                               -034-
1225 IFJV=4THENSP=2:X=X-15                              -133-
1230 IFJV=8THENSP=3:X=X+15                              -132-
1235 IFJV<>4ANDJV<>8THENSP=1                            -045-
1240 IFPX=1ANDX>255THENPOKEV+16,255:X=X-255:PX=2:GOTO
1300                                                    -023-
1245 IFPX=2ANDX<0THENPOKEV+16,0:X=X+255:PX=1:GOTO1300
                                                        -064-
1246 IFPX=2ANDX>65THENX=65:GOTO1300                     -205-
1250 IFPX=2ANDX<0THENPOKEV+16,0:X=X+255:PX=1:GOTO1300
                                                        -060-
1260 IFPX=1ANDX<25THENX=25                              -135-
1270 IFPX=2ANDX>65THENX=65:GOTO1300                     -202-
1280 IFPX=1ANDX<25THENX=25                              -137-
```

```
1300 IFVE<0THENVE=0                                    -177-
1310 IFVE>400THENVE=400                                -124-
1315 IFPEEK(V+31)=X2(SP)THEN1500                       -132-
1320 FORT=1TOVE:NEXTT:PRINTH$                          -205-
1330 IFVC=351THEN1350                                  -221-
1340 NEXTVC                                            -192-
1345 :                                                 -039-
1348 :                                                 -042-
1350 R$=TI$:PRINTH$,"{BLU}{3 DCH}"B$(10)               -142-
1360 IFPX=2THENPX=1:POKEV+16,0:X=255                   -237-
1370 PS=INT((165-X)/7):IFABS(X-165)<10THEN1460
                                                       -135-
1380 FORXS=XTO165STEPPS                                -049-
1390 IFSGN(PS)=-1THENPOKEV+21,12:SP=2:XX=4             -107-
1400 IFSGN(PS)=1THENPOKEV+21,48:SP=3:XX=8              -068-
1410 IFSGN(PS)=0THENPOKEV+21,3:SP=1:XX=0               -001-
1420 PRINTH$:POKEV+XX,XS:POKEV+XX+2,XS                 -031-
1430 NEXTXS                                            -210-
1440 POKEV,XS-PS:POKEV+2,XS-PS:POKEV+21,3:IFGG=1THENR
ETURN                                                  -088-
1450 GOTO1470                                          -239-
1460 POKEV+21,3:FORQ5=1TO8:PRINTH$:NEXT                -249-
1470 TE=(VAL(MID$(R$,3,2))):SE=VAL(RIGHT$(R$,2))
                                                       -211-
1472 RT=60*VAL(MID$(TI$,3,2))+VAL(LEFT$(TI$,2))
                                                       -128-
1475 PRINT"{HOM}{15 ABJ}{5 DCH}{RED}";                 -188-
1480 PRINT"TIEMPO";TE;"MINUTOS Y";SE;"SEGUNDOS"
                                                       -065-
1485 PU(Q2)=INT(500*VC-RT*50)                          -158-
1490 PRINT"{2 ABJ}{5 DCH}{YEL}CALIFICACION:";PU(Q2);"
PTOS."                                                 -070-
1492 IFP>Q2THENPRINT"{ABJ}{5 DCH}{GRN}GET READY PLAYE
R";Q2+1                                                -204-
1493 FORT=1TO5000:NEXT:POKEV+21,0:POKEV+16,0:POKEV,16
5:POKEV+2,165:SP=1:PX=1                                -215-
1494 IFP>Q2THENNEXTQ2                                  -131-
1496 PR=2:FORJ=12672TO12734:READA:POKEJ,A:NEXT:POKEV+
45,2:POKEV+46,2                                        -230-
1497 FORJ=12736TO12798:READA:POKEJ,A:NEXT              -068-
1498 FORPK=1TO25:READPS(PK):NEXT:KL=25:Q2=0:POKE53280
,6                                                     -186-
1499 GOTO1750                                          -253-
1500 :                                                 -032-
1501 :                                                 -033-
1540 B$=TI$:RT=VAL(MID$(B$,3,2))+VAL(RIGHT$(B$,2))
                                                       -039-
1550 POKEV+21,3                                        -093-
1560 POKEV,X:POKEV+2,X                                 -149-
1570 FORY3=Y1TO200STEP2                                -202-
1580 POKEV+1,Y3:POKEV+3,Y3+21                          -234-
1590 NEXTY3:POKEV+13,200:POKEV+15,221:POKEV+21,192:PO
KEV+12,X:POKEV+14,X                                    -144-
1600 X=165:POKEV+1,107:POKEV+3,128:DU=10:GOSUB2200
                                                       -202-
1610 PRINT"{CLR}"                                      -076-
1650 PRINTTAB(5);"{2 ABJ}TIEMPO: ";MID$(B$,3,2)" MINU
TOS Y "RIGHT$(B$,2)" SEGUNDOS{ABJ}"                    -012-
1670 PU(Q2)=VC*100-RT*50                               -090-
1680 PRINTTAB(5);"PUNTUACION:";PU(Q2);"PUNTOS"
                                                       -180-
1700 GOTO1492                                          -241-
1720 :                                                 -036-
1730 :                                                 -037-
1750 PRINT"{CLR}":IY=0:DU=1                            -219-
1751 XC=0:PX=1:GG=1:PQ=1:POKEV+12,PS(PQ):Q2=Q2+1:IFL=
1THENTP$="5"                                           -220-
1752 POKEV+1,117:POKEV+3,138:POKEV+5,117:POKEV+7,138:
POKEV+9,117:POKEV+11,138                               -067-
1755 Y1=255:Y2=255:U=0:FF=192:POKEV+15,255:TI$="[6 0]
":YH=0:FA=-2:RA=7                                      -040-
1757 IFL=1THENTP$="5"                                  -234-
1758 IFL=2THENTP$="3"                                  -234-
1760 GOSUB1007                                         -054-
1770 Y1=Y1-RA:IFY1<=0THENY1=255:U1=0:GOSUB2030
                                                       -058-
1780 POKEV+13,Y1                                       -186-
1785 IFY1<132THENU=1                                   -210-
1790 IFU=0THEN1820                                     -060-
1800 Y2=Y2-RA:IFY2<=0THENY2=255:U2=0:GOSUB2000
                                                       -054-
1810 POKEV+15,Y2                                       -183-
1815 IFY1<132ANDU1=0THENU1=1:GOTO2300                  -251-
1820 IFY2<132ANDU2=0THENU2=1:GOTO2330                  -253-
1822 IFPEEK(V+30)=64+X2(SP)THENPOKEV+45,0:CK=CK+1:GOS
UB2200                                                 -055-
1830 JV=PEEK(56320):FR=JVAND16                         -040-
1840 JV=15-(JVAND15)                                   -135-
1850 IFJV=4THENSP=2:X=X-15:XT=X:GOTO1870               -071-
1860 IFJV=8THENSP=3:X=X+15:XT=X:GOTO1870               -075-
1865 SP=1                                              -005-
1870 IFFR<>16THENRA=RA+5:IFRA>15THENRA=15:FY=1
                                                       -150-
1871 IFX>255THENPX=2:POKEV+16,63:XT=X:X=X-255
                                                       -073-
1872 IFX<0ANDPX=2THENPX=1:X=X+255:POKEV+16,0:XT=X
                                                       -138-
1873 IFX<25ANDPX=1THENX=25                             -145-
1874 IFX>65ANDPX=2THENX=65                             -157-
1875 IFPX=2THENXT=X+255                                -210-
1889 FA$=STR$(FA)                                      -215-
1890 IFFY=0THENRA=RA-1:IFRA=6THENRA=7                  -130-
1893 MI$=MID$(TI$,3,2)                                 -185-
1894 SG$=RIGHT$(TI$,2)                                 -003-
1900 PRINT"{HOM}{BLU}"TAB(30)MI$":"SG$                 -110-
1910 FY=0:GOTO1760                                     -056-
1920 :                                                 -038-
1930 :                                                 -039-
2000 PQ=PQ+1:POKEV+14,PS(PQ)                           -205-
2020 GOTO2040                                          -227-
2030 PQ=PQ+1:POKEV+12,PS(PQ)                           -206-
2040 IFPQ=25THEN2080                                   -179-
2050 POKEV+45,2:POKEV+46,2                             -016-
2070 RETURN                                            -201-
2080 GOSUB1350                                         -051-
2085 PRINT"{CLR}{9 ABJ}"                               -236-
2086 PRINT"{2 ABJ}{BLU}"TAB(5)"TIEMPO: ";MI$" MINUTOS
 Y ";SG$" SEGUNDOS"                                    -177-
2087 TJ=VAL(MI$)*60+VAL(SG$):PU(Q2)=PU(Q2)+(50000-100
*TJ)-CK*500                                            -065-
2088 PRINT"{2 ABJ}"TAB(5)"{RED}PUNTUACION PRUEBA 2:";
(50000-100*TJ)-CK*500;"PUNTOS                          -218-
2089 PRINT"{2 ABJ}"TAB(5)"{BLK}PUNTUACION TOTAL:";PU(
Q2);"PUNTOS                                            -245-
2090 FORT=1TO5000:NEXT                                 -117-
2092 IFP>Q2THENFA=0:U=0:U1=0:U2=0:FORT=1TO2500:NEXT:G
OTO1750                                                -013-
2095 GOTO2500                                          -240-
2110 GOTO2110                                          -225-
2200 FORS=54272TO54296:POKES,0:NEXT:POKE54296,15
                                                       -162-
2202 FORT2=1TODU:POKE54273,17:POKE54272,37             -002-
2205 POKE54278,10:POKE54276,129:POKE54278,120:POKE542
77,9                                                   -241-
2220 NEXTT2                                            -171-
2225 POKE54296,0                                       -128-
2230 RETURN                                            -199-
2300 IFPQ=1ANDXT<190THENFA=FA+1:GOTO2350               -235-
2301 IFPQ=1THEN1822                                    -128-
2302 IFPEEK(V+12)<XTTHEN2350                           -177-
2320 GOTO1820                                          -235-
2330 IFPEEK(V+14)>XTTHEN2350                           -182-
2340 GOTO2380                                          -239-
2350 FA=FA+1                                           -145-
2370 IFFA=VAL(TP$)THEN2410                             -049-
2380 GOTO1822                                          -243-
2410 PRINT"{HOM}{16 ABJ}"TAB(5)"ELIMINADO"             -014-
2420 PS=PQ*500+VAL(TI$)*50                             -185-
2430 PRINT"{2 ABJ}"TAB(5)"PUNTUACION: ";PS;"PUNTOS"
                                                       -005-
2440 PU(Q2)=PU(Q2)+PS                                  -201-
2450 PRINT"{2 ABJ}"TAB(5)"{RED}PUNTUACION TOTAL:";PU(
Q2);"PUNTOS"                                           -155-
2460 FORT=1TO2500:NEXT:GOTO2090                        -182-
2480 :                                                 -040-
2490 :                                                 -041-
2500 :                                                 -033-
2510 PRINT"{CLR}{2 ABJ}{BLK}":POKEV+21,0               -167-
```

```
2590 RF=0:FORA=1TOP:IFPU(A)>PI(5)THENGOSUB2600:RF=1
                                                     -204-
2595 NEXTA:IFRF=0THEN2700                            -059-
2597 GOTO2625                                        -255-
2600 PRINT"{CLR}{5 ABJ}","NOMBRE JUGADOR";A;:INPUTPI$
(5):PI(5)=PU(A)                                      -145-
2608 FORE=1TO4:FORE2=1TO4                            -062-
2610 IFPI(E2)<=PI(E2+1)THEN2620                      -008-
2615 X=PI(E2):PI(E2)=PI(E2+2):PI(E2+1)=X             -006-
2617 X$=PI$(E2):PI$(E2)=PI$(E2+1):PI$(E2)=X$
                                                     -131-
2620 NEXTE2:NEXTE                                    -094-
2625 PRINT"{CLR}":FORT=1TO10:FORTE=1TO7              -060-
2630 PRINTCO$(TE)"{HOM}{3 ABJ}"TAB(15)"RECORDS"
                                                     -134-
2640 PRINTTAB(15)"¦7 *¦{ABJ}"                        -156-
2650 FORAF=5TO1STEP-1                                -059-
2660 PRINT"{2 ABJ}",LEFT$(PI$(AF),8);TAB(22)PI(AF)"PT
OS."                                                 -150-
2670 NEXTAF:NEXTTE:NEXTT                             -148-
2700 PRINT"{CLR}{11 ABJ}"TAB(7)"DESEA JUGAR OTRA VEZ
(S/N)"                                               -082-
2710 GETA$:IFA$="S"THEN50                            -197-
2720 IFA$<>"N"THEN2710                               -228-
2730 PRINT"{CLR}"                                    -080-
2740 END                                             -196-
29900 :                                              -094-
29910 :                                              -095-
29940 DATA"{RED}","{CYN}","{GRN}","{YEL}","{PUR}","{B
LK}","{ACL}"                                         -099-
30000 DATA0,40,0,0,170,0,0,150,0,0,150,0,0,150,0,0,20
,0,0,20,0,10,170,160                                 -211-
30010 DATA42,170,168,170,170,170,170,170,170,162,170,
138,162,170,138                                      -093-
30020 DATA162,170,138,162,170,138,162,170,138,162,170
,138,162,170,138                                     -152-
30030 DATA162,170,138,162,170,138,162,170,138
                                                     -247-
30040 DATA130,170,130,130,130,130,130,130,130,194,130
,131,194,130,131                                     -095-
30050 DATA194,130,131,194,130,131,194,130,131,194,130
,131,194,130,131                                     -125-
30060 DATA194,130,131,194,130,131,194,130,131,195,195
,195,195,195,195                                     -170-
30070 DATA195,195,195,3,195,192,3,195,192,3,195,192,3
,195,192,3,195,192                                   -035-
30090 DATA0,0,0,0,10,128,0,10,128,0,5,128,0,5,128,0,5
,128,0,5,128,0,170,160                               -113-
30100 DATA2,170,168,2,170,168,2,42,168,2,42,136,2,42,
136,10,42,136,10,42,136                              -205-
30110 DATA8,42,8,8,42,8,40,42,8,40,42,40,40,170,40,32
,168,32,32,168,32                                    -173-
30130 DATA160,136,160,128,136,128,192,136,192,240,136
,240,60,136,60,15,136,15                             -025-
30140 DATA3,136,3,0,136,0,0,136,192,0,139,192,0,139,0
,2,184,48,2,248,240                                  -009-
30150 DATA3,235,240,15,47,0,60,60,0,240,240,0,3,192,0
,7,0,0,60,0,0,0,0,0                                  -207-
30170 DATA0,0,0,2,160,0,2,160,0,2,80,0,2,80,0,2,80,0,
2,80,0,10,170,0                                      -252-
30180 DATA42,170,128,42,170,128,42,168,128,34,168,128
,34,168,128,34,168,160                               -211-
30190 DATA32,168,32,32,168,32,40,168,32,40,170,40,8,4
2,8,8,42,8                                           -105-
30210 DATA10,34,10,2,34,2,3,34,3,15,34,15,60,34,60,24
0,34,240,192,34,192                                  -249-
30220 DATA0,34,0,0,34,0,0,226,0,0,34,0,0,46,128,15,67
,128,3,235,192,0,251,240                             -222-
30230 DATA0,60,60,0,15,15,0,3,195,0,0,240,0,0,60,0,0,
15                                                   -154-
30250 DATA0,0,0,0,0,0,0,0,0,0,0,0,0                  -108-
30260 DATA0,240,0,3,192,0,255,0,0,60,0,0,255,0,3,195,
192,15,0,240,0,0,60                                  -224-
30270 DATA0,0,15,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,
0,0,0,0                                              -060-
30290 DATA0,0,0,0,0,0,0,0,0,0,0                      -092-
30300 DATA84,0,0,84,0,0,232,9,0,232,41,0,232,160,10,4
2,170,170,10,168,0                                   -185-
30310 DATA10,170,170,8,0,10,10,0,0,2,128,0,0,164,0,0,
36,0                                                 -249-
30320 DATA0,0,0,0,0,0,0,0,0,0,0                      -086-
30340 DATA128,0,0,160,0,0,168,0,0,170,0,0,170,128,0,1
70,160,0,170,168,0                                   -182-
30350 DATA170,170,0,170,170,128                      -060-
30360 DATA160,0,0,160,0,0,160,0,0,160,0,0,160,0,0,160
,0,0                                                 -236-
30370 DATA160,0,0,160,0,0,160,0,0,160,0,0,160,0,0,160
,0,0                                                 -237-
30390 DATA0,0,2,0,0,10,0,0,42,0,0,170,0,2,170,0,10,17
0,0,42,170,0,170,170                                 -239-
30400 DATA2,170,170                                  -235-
30410 DATA0,0,10,0,0,10,0,0,10,0,0,10,0,0,10,0,0,10
                                                     -164-
30420 DATA0,0,10,0,0,10,0,0,10,0,0,10,0,0,10,0,0,10
                                                     -165-
30440 DATA190,150,175,155,210,140,220,135,230,100,240
,90,250,80,220,100                                   -207-
30450 DATA200,130,175,140,190,125,175,125,155,135,150
                                                     -119-

READY.                                               -173-
```



CHISPA:

En los ordenadores Commodore se pueden eliminar los espacios del programa en BASIC; esto resulta cómodo para el usuario, pues es el intérprete el que descifra las palabras clave que van apareciendo en forma de «tokens» (una palabra = un byte), diferenciándolas de variables, texto, etc. Es por eso por lo que hay que tener cuidado con lo que llamaremos «tokenización escondida»; por ejemplo, la siguiente sintaxis es correcta:

IF HT AND P THEN...

Que indica que si la variable ht y la variable p son distintas de cero entonces...

Pero veamos lo que ocurre si en esa construcción suprimimos los espacios:

IFHTANDPTHEN...

Ahora el intérprete generará un SYNTAX ERROR?, ya que lo que descifrará será: Si la variable h es distinta de cero tangente de dp... Lo cual es absurdo.

Lo que ha ocurrido es que sin querer se ha formado una palabra clave (TAN) del BASIC, al unir la última letra de la variable ht (T) y las dos primeras de la palabra and (AN). Debemos tener especial cuidado en evitar este tipo de casualidades. La sintaxis correcta sería:

IFHT ANDPTHEN...

Separando con un espacio la zona donde puede originarse la confusión.

CHISPA:

Hay veces en las que, al operar con la unidad de disco, cometemos el error, generalmente involuntario, de grabar como fichero o como programa, algo con el nombre (*) o (,). Al ser estos caracteres tratados de una forma especial por el DOS, como códigos prohibidos dentro del nombre de ficheros o (programas), a la hora de borrarlos o cambiarlos de nombre surgen los problemas, ya que todo intento de eliminarlos por la vía habitual, será fallido. Para ello deberemos proceder de la siguiente manera:

OPEN 15,8,15,"SO:?".

Este comando borrará todos los ficheros presentes en el disco, cuyo nombre conste sólo de una letra: así pues, si deseamos conservar algún fichero de estas características, habremos de grabarlo en otro disco (o en el mismo con otro nombre de más de una letra, si cabe), antes de proceder a su uso.

CHISPA:

Para efectuar operaciones con el valor de una variable depositado en memoria, lo primero que hay que saber es en qué zona comienza dicha variable. Este trabajo necesita una comprensión extensa de la forma en que se almacenan las variables en memoria. Como ya imagináis, hay una rutina en ROM que nos devuelve la posición de inicio (esta es utilizada por el intérprete). Observemos la siguiente línea de programa y veremos como trabaja:

```
10 A=123:P=0:
SYS45195A:P=PEEK(780)+
256*PEEK(782):PRINTP
```

En primer lugar, hay que definir la variable a estudiar (A) y todas las que se vayan a emplear como ayuda (P). A continuación se efectúa un SYS a la dirección 45195, seguido del nombre de la variable a examinar, sin separación alguna. La dirección de la variable será devuelta en forma byte bajo-byte alto en el acumulador (780) y en el registro Y (782). La línea anterior escribirá la dirección. Si preguntamos por el valor de la dirección menos dos bytes, obtendremos el código ASCII del primer carácter del nombre de la variable.

CHISPA:

En los programas escritos en código máquina, el equivalente de RUN en BASIC es un SYS a una determinada dirección. Normalmente, esta dirección coincide con el inicio del programa. La siguiente rutina para programas en disco, puede darnos en poco tiempo la información necesaria del inicio del programa, para aquellos casos en que no sepamos cuál es. Los usuarios de cinta, deberán cambiar el OPEN 1, 8, 8... por OPEN 1, 1, 1... en la línea 20.

```
10 INPUT "NOMBRE DEL
   PROGRAMA";N$
20 OPEN1,8,8,N$+",P,R":-
   GET#1,A$:GET#1,B$
30 PRINT"DIRECCION DE
   INICIO:";ASC(A$+
   CHR$(0))+
   256*ASC(B$+
   CHR$(0)):CLOSE1
```

CHISPA:

Muchos de nosotros nos hemos preguntado alguna vez, cómo es que C64 se anuncia como un ordenador de 64K RAM si al conectar nos dice el mensaje de inicio que tenemos 38K libres.

Bien, para explicar esto hay que saber que todo lo que al encender es ROM, se puede convertir en RAM (perdiendo momentáneamente el acceso a dicha ROM). Para efectuar estos cambios está la posición de memoria 1.

Para comprobarlo, pondremos en práctica el siguiente ejemplo, cuya primera misión es tomar el área del intérprete BASIC y ¡escribid lo mismo que ha leído! en toda su zona de memoria. Esta operación sirve para leer el contenido de la ROM y escribirlo en la RAM que está DEBAJO de esta ROM; dichos datos se perderían (la de la RAM), a no ser que le dijéramos al C64 que buscara el intérprete BASIC en la RAM de DEBAJO con POKE1,PEEK(1)AND254.

Ahora tenemos el intérprete BASIC a nuestro alcance (¡Cuidado cualquier POKE erróneo en esta zona «colgará» al ordenador!). En este ejemplo modificamos la palabra READY. con los seis caracteres que le metamos en la variable R$ de la línea 20.

```
10 FORI=40960TO49151:
   POKEI,PEEK(I):
   NEXT:POKE1,
   PEEK(1)AND254
20
R$;"LISTO.":FORI=
1TOLEN(R$):POKE41847+
I,ASC(MID$(R$,I,1)):NEXT
```

CHISPA:

Para aquellos propietarios de un C64 con unidad de disco 1541, que quieran «oír» la carga desde el disco, va la siguiente chispa. Con el volumen de la televisión o monitor bastante alto y efectuando **POKE54296,15**, procederemos a cargar algo desde el disco; oiremos la carga por bloques. No es muy rápido, ¿verdad?

CHISPA:

En los ordenadores Commodore que disponen de un botón de RESET (C16, PLUS 4 y C128), éste es útil para salir del atolladero cuando el ordenador se «cuelga» o está en «lockup» (en el idioma que ya sabemos); pero este RESET acaba con cualquier programa BASIC que tengamos en ese momento en memoria. Para evitar esto, bastará con mantener pulsada la tecla RUN/STOP mientras se actúa sobre el referido botón; con ello conseguiremos entrar en el monitor de código máquina interno que tienen estos ordenadores, pero sin perder el programa BASIC, bastando teclear X y RETURN para volver al BASIC sin perder el programa que hubiese antes del fatídico «lockup».

PARA LLEGAR AL MUNDO DE LOS ORDENADORES PERSONALES/PROFESIONALES.

EL MEJOR MEDIO PARA MEJORAR SU IMAGEN PERSONAL.



Usuarios actuales y potenciales, especialistas en marketing, responsables comerciales, técnicos... etc. TODOS NOS LEEN.

Y también leerán sus anuncios, porque TU MICRO PERSONAL informa con agilidad y amenidad sobre la realidad del mercado de los ordenadores personales/profesionales, rentabilizando la inversión y aumentando la productividad de los P.C., por lo tanto interesa no sólo a los usuarios sino que es buscada y leída por los responsables comerciales y de marketing de este importante sector.

SU IMAGEN CON CALIDAD.

En TU MICRO PERSONAL tenemos una buena imagen, cuidamos la reproducción para que gráficamente su publicidad responda a la imagen con que fue creada.

TU MICRO PERSONAL.

Tenemos buena imagen.



Plza. República del Ecuador, 2-1º-28016 Madrid



Os recodarmos nuevamente, que ninguna carta con ánimo de lucro o que huela tan sólo ligeramente a «pirata», olor fétido de descomposición cerebral fácilmente reconocible, será publicada en nuestras páginas.

Además, aprovechamos la ocasión para haceros un ruego: escribid el texto de vuestros anuncios en un papel a parte de cualquier otro que enviéis con sugerencias, etc., y emplead letra de imprenta lo más clara que podáis, a máquina siempre que os sea posible.

INTERCAMBIO software para C64, poseo interesantes novedades. Vendo Viopac Philips G-7400, regalo cartuchos y 300 programas para C64 de última novedad todo por sólo 10.000 ptas. (el Videopac es convertible en ordenador de 16K, ideal para los que empiezan). Busco unidad de disco barata. Francisco Javier Bernal Malillos, Echegaray, 1. Cantalejo (Segovia). Tel. (911) 520404.

CAMBIO programas para C64. Interesados llamar al Tel.: (91) 213 38 76. Preguntar por Marcos Sanz Gallego. Carretera de Vicálvaro, 132, 4.° D. 28022 Madrid.

COMPRO vendo o cambio programas para C64, también estoy interesado en comprar un joystick, escribir a: Gonzalo Carvajal Berbel. Sant Medir, 12-14, 2.° 1.ª Sant Cugat del Valles. 08190 Barcelona.

INTERCAMBIAMOS Programas para C64/128, en disco o cinta. Contestamos a todas las cartas. Club Commodore Canario. Apartado 2.485. Las Palmas de Gran Canaria.

INTERCAMBIO programas en cinta para C64. Poseo las últimas novedades (Silent Service, Leader Board, Harball, etc.). Interesados mandar lista a: Andrés Fernández Morlanes. General Yague, 25. 41500 Alcalá de Guadaria (Sevilla).

VENDO monitor en color marca Philips (sólo 4 meses de uso) por sólo 30.000 ptas. Dirigirse a Cristina. Barcelona. Tel.: (93) 203 21 64.

CAMBIO programas para C-64. Escribir a Joan Rimbas. Lapant, 5-5.° 1.ª 43202 Reus (Barcelona).

VENDO C64 comprado en septiembre del 86 maś dos libros de instrucciones y programas más unidad de cassette, más amplia gama de juegos de la categoría de Rambo, commando, The Way of the Tiger, Exploding Fist, Pitstop II, Ghost Busters y muchos más por tan sólo 40.000 ptas. Interesados llamar o escribir a Aleix Oril. Vizcaya, 346, 7.° J. 08027 Barcelona. Tel.: (93) 349 06 28.

COMPRO Datassete C2N, que esté en buenas condiciones. También cambio programas para C64 en cinta. Enviad vuestra lista. Interesados escribid a: Antonio Pelegrín Cruz. Fdco. Echevarría, 1, 6.° 24002 León. Tel.: (987) 23 51 31 de 19 a 24 h. Prometo contestar.

VENDO C64, Datassete, teclado de órgano acoplable al ordenador (con su correspondiente programa), regalo joystick y algunos programas, todo por 45.000 ptas. Dirigirse a Felipe S. Herrero Rguez. Jacinto Borges, 13. 35500 Arrecife de Lanzarote.

CLUB USUARIOS Commodore-Tarragona, centro homologado por la Generalitat de Catalunya; nuestras actividades están distribuidas en distintas secciones: Biblioteca, programoteca, cursos, asesoramiento, etc. Dos años al servicio de nuestros socios. Para información: Club Usuarios Commodore. Apartado de Correos 176. Tarragona. O también: Fortuny, 4, 2.° 2.ª Además cursos intensivos de Basic. ¡Infórmate!

CAMBIO programas de la calidad de Uridium, Ghost'n Goblins, V, Bomb Jack, Nexus, Saboteur, etc. por hardware CBM o compatible, preferiblemente 1541. Cantidad a convenir. También intercambio programas para C64. Escribid o llamar a: Marcos Miguel Huerga Muñoz. Candelaria Ruiz del Arbol, 55, 4.° 3 A. 49003 Zamora. Tel.: (988) 52 88 64.

INTERCAMBIO utilidades y programas en cinta para C64. Prometo contestar todas las cartas. Mi dirección es: María José Villamayor. Aden, 3 bajos. 17220 Sant Feliu de Guixol (Gerona).

INTERCAMBIO juegos de C64. Interesados contactar con Sergio Frías. Balmes, 306, 3.°. 08006 Barcelona. Tel.: (93) 209 76 62. Preferiblemente llamar por teléfono.

VENDO Cartucho Simon's Basic (6.000 ptas.), cartuchos Internacional Soccer y Rotting Toting (2.000 ptas. unidad). Todos son originales, nuevos y con instrucciones. José Ramón Diaz Glez. Río Requejada, 2, 1 izq. 33460 Llanares. Avilés (Asturias).

DESEARIA intercambiar juegos para C64. Tengo algunos muy buenos. Mandad lista. Prometo contestar a todas las cartas. Interesados escribid a: Pedro Fuertes Penella. Duque de Mandas, 27, 1.° 46019 Valencia.

CLUB COMMODORE de Alcañiz. Nos gustaría contactar con colegas españoles. Intercambio de juegos y utilidades. Tenemos muchos y buenos. Dirigirse al Apartado 49. Alcañiz (Teruel). O al Tel.: 83 04 56.

VENDO C-64 (44.000 ptas.) y datassette Commodore (6.000 ptas.) juntos o por separado (con instrucciones y conexiones). Sólo en Barcelona. Tel.: 334 14 14 de 19 a 21h.

INTERCAMBIO programas de todo tipo para C64, sobre todo de música y utilidades, poseo disco y cassette. Mandad lista poseo más de 500 programas. Francisco Ballesteros. Apartado 120. 03080 Santa Pola (Alicante).

VENDO impresora Commodore MPS-801, comprada a Microelectrónica y Control, casi nueva. Llamar a Magdalena (sólo por la mañana) al Tel.: (91) 213 13 78.

ME GUSTARIA contactar con usuarios del C-64, así como con clubes para el intercambio de programas, trucos, experiencias, etc. Escribid de toda España. Tengo buenos programas. Me interesa unidad de disco y revistas de toda clase para Commodore, todo ello en perfecto estado y barato. Compraría copión para duplicar mis programas. Mi dirección es: Chris Márquez Sánchez. Cuenca, 11, 6.°. Cantarroja (Valencia).

CAMBIO vídeo-juego Atari con manual y cables en perfecto estado y con varios cartuchos de juegos, valorado en 50.000 ptas., por unidad de disco de segunda mano para Commodore. Mis señas son: Chris Márquez Sánchez. Cuenca, 11, 6.°. Cantarroja (Valencia).

INTERCAMBIO programas para C-64. Prometo contestar. Mandar lista a: Toni Altadill Mora. Bon Aire, 6. 43500 Tortosa.

INTERCAMBIO programas de todo tipo para C128 preferentemente en disco. Enviar lista a: Marco Rossi. Masferrer, 14, 1.° 08028 Barcelona. Tel.: 330 8780.

SOLO EN Barcelona. Vendo las siguiente revistas: Commodore World n.os 2, 5, 6, 8, 9, 11, 12, 13, 14, 15, 16 y 17 a 200 ptas. cada una; Commodore Magazine del 1 al 233, menos el 16 a 200 ptas. cada una; Load'n' run n.os 1, 3 y 5 a 750 ptas. cada uno, y el n.° 5 extra de verano a 1.000 ptas. Los interesados ponerse en contacto con Marcos Recacha Fontana. Dr. Pí y Molist, 172, ent. 2.°. 08031 Barcelona. Al escribir poned vuestro teléfono. Cambio juegos, poseo 300, muy buenos y todos en turbo (Commando, The Last V8, Kunf-Fu Master).

NECESITO recaudar fondos para la compra de un C-128, por lo que vendo el siguiente material para el C-64: 3 cartuchos (Socer, Simons' Basic, Hesware y 200 juegos tales como Rambo, Comman-

do y utilidades como Simulador de Spectrum, Busucalc, Compositor, etc. También vendo las siguiente revistas: Nuevas Tecnologías n.º 1, Biblioteca Básica de Informática n.º, Lenguaje Máquina del C-64, Muy ordenadores n.os 4, 5 y 6, Commodore World n.os 17, 19, del 20 al 26 y 28, Commodore Magazine n.os del 12 al 23 y 28, Basic n.º 121, Input Commodore del 1 al 6 y 8, Mi Computer n.os 21, 23 y 25, Ordenador personal n.os 22, 24 y 41. Todo aquel que tenga interés por cualquiera de estas cosas, se dirija a: Miguel Angel Huelves García. Entre arroyos, 64. 28030 Madrid.

INTERCAMBIO programas para C-64 exclusivamente en cinta. Mi dirección es: José Pérez Mesas. Segundo paseo de Ronda, 47, 4.º, 3.ª. 25008 Lérida. Tel.: (973) 24 62 88. Telefoneadme o escribidme, prometo contestaros a todos.

DESEARIAMOS formar un club de usuarios del CBM 64 en Avilés, para intercambiar programas y experiencias. Interesados llamar a Santi al Tel.: 57 45 38 o escribir a Luis Santiago Ruiz Seijo. Pza. Vistahermosa, 9, 2.º D. 33400 Corvera (Asturias).

SOLO EN Orense. Intercambiaría todo tipo de juegos para C-64 o cambiaría Hyper Sports por cualquier otro programa, precio 1.000 ptas. Interesados escribir a Emilio López Marino. Samuel Eiján, 5, 1.º B. Orense. Prometo contestar.

INTERCAMBIO programas en cinta y disco para C-64. Interesados mandar listas a Fernando Lanchares González. Ronde del Carmen, 25, 1.º B. 10002 Cáceres.

DESEARIAMOS crear un club de usuarios de ordenador de Orense, para intercambiar ideas, información, etc. Interesados dirigirse a Curro Enríquez, 73 (al lado de Judo Club Lee) en Orense.

INTERCAMBIO programas en cinta para C-64. Interesados enviar lista a Carlos Borreguero Peralbo. Princesa, 4, 1.º A. Mora d'Ebre (Tarragona). Tel.: (977) 40 18 99.

CAMBIO dos joysticks + 25 revistas + 2 libros de programación + Interface Copiador + 700 programas (úlitmos: Silent Service, Leader Board, Ghost'n Gobblins, etc.) con sus instrucciones, por unidad de disco 1541 en buen estado. También cambio programas. Benjamín Blázquez Calzada. S. Juan Salle, 63. 41008 Sevilla. Tel.: (954) 43 36 39.

VENDO cartucho The Final Cartridge, en perfecto estado por 9.000 ptas. (lo vendo por la compra del nuevo modelo). José del Aguila. Tel.: (93) 372 80 71.

AMIGOS commodorianos: cambio programas en soporte cinta, interesados enviar lista a Doctor Fleming, 19. 25006 Lérida.

COMPRO Datassette C2N, que esté en perfecto estado y a buen precio. También estoy interesado en el intercambio de programas, preferentemente en soporte disco. Carlos de la Presa Gómnez. Montepinar, 8. Guadarrama. 28440 Madrid. Tel.: (91) 854 00 33.

C-64. Interesado en las comunicaciones busca otros usuarios interesados en el mismo tema, y también para el intercambio de software. Escribid o llamar a Eduard Macip. Forn de lasa vila, 4. 42400 Montblanc (Tarragona). Tel.: (977) 86 00 63. (Lunes a viernes).

VENDO Commodore Vic-20 más instrucciones y libros de iniciación por 9.900 ptas. (sin gasto administrativo alguno), en perfecto estado (adquirido en tienda con 9 meses de uso). Interesados llamar a José Luis. Tel.: (965) 49 11 37 de Aspe.

INTERCAMBIO programas para C-64, tanto juegos como utilidades. Enviar lista, prometo contestar. Antonio Molina García. Barriada de la Constitución, Bloque 12, 2.º C. Melilla.

CAMBIO programas por hardware de cualquier tipo, y vendo programas originales (no copias) a 1.550 y 2.950 ptas. cada uno. Llamar al Tel.: (91) 472 03 99, preguntar por Juan.

CAMBIO programas de todo tipo para C-64, en cinta o disco. Mandar lista, contestaré a todas las cartas. Mi dirección Luis Rivira Roig. Avda. Madrid, 106. Entlo. 2.ª. 08028 Barcelona.

VENDO C64 completamente nuevo (garantía 4 meses), más datassette, más 25 juegos, más curso de informática. Precio a convenir. Joan Ricard Solá i Godoy. Mare de Port, 375-Ato 2.º 08004 Barcelona. Tel.: (93) 331 83 76.

VENDO e intercambio programas del C64. Todos a buen precio, mandadme lista, contestaré. Vendo programas como: Speed King, Archon, Simon's Basic. En cinta. Escribid a: Carlos J. Amez Lemaur, bloq. Chile, 8-10, 1.º S. Pedro S. Pablo. 43007 Tarragona, o llamar al Tel.: (977) 22 10 93.

VENDO Vic-20 en ampliación de memoria 68K, curso de basic y juegos, así como el cassette, todo muy económico. Llamar martes, jueves y viernes por la mañana a Juan al Tel.: (93) 340 38 99 de Barcelona.

COMPRO ampliación de memoria para Vic-20, a ser posible 16K. Preguntar por Jordi Gómez Raya al Tel.: (93) 762 52 28. Ausias March, 13. Pineda del Mar. 08397 Barcelona.

SE HACEN todo tipo de trabajos con impresora, como listados (desde casete o disco), etiquetas para mailing (con programa destinado a ello). Todo con una impresora MPS-801, mejorada con un chip especial. A toda España, muy barato. Dirigirse a Gerardo Rodríguez Aliaga. Viviendas obras públicas, 13. 29740 Torre del Mar (Málaga). Contestación asegurada.

VENDO por cambio de equipo: Commodore 128 con datassete en buen estado así como más de 30 juegos de los mejores del mercado nacional e internacional (Drop Zone, The Dam Busters, Doriath, Castle, Winter Games, Summer Games I y II, Ballblazer, Strangeloop, Bruce Lee, Leader Board Golf, etc.). Todo por 75.000 ptas. Interesados ponerse en contacto con: Antonio Rodríguez Bueno. Mallorca, s/n. San Fernando. Formentera (Baleares). Tel.: (971) 32 02 99, llamar a partir de las 20h.

VENDO Commodore 64, Incluyo: Datassette y varios juegos. Precio 33.000 ptas. Interesados llamar por la noche al Tel.: 258 87 46 de Barcelona.

QUISIERA poder comunicarme con personas interesadas en el intercambio de juegos y utilidades para C64, además de formar una duradera amistad intercontinental. Interesados escribir a Franco D'Agostino. Vicente López, 1756. Moron. Buenos Aires (Argentina). Código Postal 1708.

POSEO C64 con casete, bastantes juegos, copiones turbo, etc. Enviar lista, prometo contestar todas las cartas, Interesados escribir a: Paco Molina Martínez. Torres Quevedo, 76, 1.º izq. 02003 Albacete.

INTERCAMBIO programas para C64 en cinta, poseo unos 300 (gran cantidad de utilidades). También vendo Simon's Basic con instrucciones, o lo cambiaría por un buen procesador de textos. Mi dirección es: Fco. Javier García Escoms. Preso, 15. Carlet (Valencia). Tel.: 253 05 85.

INTERCAMBIO programas. Poseo algunos muy buenos. Mandad lista, prometo contestar. Sólo en cinta. Ignacio Sánchez Sampuyo. Ezcurdia, 95 bis-3 izq. Gijón. 33203 Asturias.

INTERCAMBIO todo tipo de programas para C64, preferentemente musicales y juegos. Enviad vuestra lista de programas. Dirigios a: Antonio Saa Quintas. Almendro, 120. 04720 Aguadulce (Almería).

INTERCAMBIO programas para Commodore 64 y 128. Sólo disco. Enviad lista. Fco. Javier Alonso Quijada. García Barbón, 104-5 puerta 7. Vigo. Pontevedra.

SILENCIO, no pierda la oportunidad de leer este anuncio, se vende por cambio de modelo superior, todo lo que detallo: lápiz óptico (5.000 ptas), cartucho de juego Matrix (4.000 ptas), cartucho Soccer (2.500 ptas.), Simon's Basic con instrucciones (6.000 ptas.), interface copiador para todo tipo de software (3.000 ptas.), Vídeo Basic (450 ptas. cada uno). Regalo con la compra de alguna de estas piezas, varios programas. Arturo Mercade. 11 de Septiembre, 1-1. 43830 Torredembarra (Tarragona).

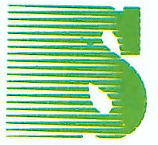
ME GUSTARIA me mandaráis instrucciones de alguno de estos juegos. Nexus, Titanic. Pagaría fotocopias y gastos de envío. Enviad a: Alberto Sire Langa. Tardirt, 7, 2.º 08025 Barcelona.

INTERCAMBIO juegos Commodore 64. Dirigirse a: Juan Manuel Aliaga Boigues. Alcacer, 39, patio 32. 46014 Valencia. Tel.: 379 81 41.

DESEARIA intercambiar programas para C64. Respondo a todas las cartas. Lorena Celeste Anguita. Esquiu, 3.336. 1879 Quilmes Oeste. Provincia de Buenos Aires. República Argentina.

C-128 y vídeo de 80 columnas

Con este montaje nuestro monitor de vídeo compuesto podrá visualizar las 80 columnas disponibles en la salida de RGBI. Gracias a ello ahorraremos una cantidad apreciable de dinero, ya que el monitor RGBI es sensiblemente más caro que el de vídeo compuesto.



Si tienes un monitor monocromo para vídeo compuesto monocromo, estás de suerte ya que en ellos es donde mayor resolución se alcanza. Con él se ven mucho mejor los caracteres del vídeo de 80 columnas; éstos, de dimensiones menores a los del vídeo de 40 columnas, se hacen difíciles de visualizar en los de vídeo compuesto de color.

Para obtener una buena imagen de 80 columnas en color tendremos que recurrir a un monitor RGBI con el desembolso que ello supone. Además debemos tener presente que en el formato de 80 columnas no se pueden visualizar gráficos de alta resolución, por lo que la necesidad del color puede ser cuestionada.

Hemos hecho pruebas con algunos monitores de color para vídeo compuesto y la resolución no es muy buena, por lo que si disponéis de uno de este tipo debéis tenerlo en cuenta a la hora de pensar en hacer este montaje.

Por el contrario, en las pruebas realizadas sobre un monitor monocromo, la visualización de los caracteres ha sido buena, aunque en el fondo esta calidad va a depender de las características del monitor.

Al principio tuvimos problemas de sincronismo al conmutar del vídeo de 40 columnas al de 80, pues la pantalla oscilaba y lo mismo ocurría en el modo 80 columnas al pulsar RUN/STOP RESTORE. Este problema tuvo fácil solución al retocar el potenciómetro de sincronismo vertical del monitor; procedimos colocando dicho potenciómetro en una posición intermedia que sirviera para los dos formatos de pantalla.

Hemos de tener en cuenta que el sincronismo del vídeo de 80 columnas es un poco más delicado que el de 40; por ello, si nuestro monitor anda un poco crítico de ajuste de sincronismo vertical, puede que no encontremos una posición intermedia para los dos tipos de formato. De ser así, debemos hacer un ajuste cada vez que decidamos conmutar de un modo a otro.

Por otra parte, la escala de verdes que hemos logrado obtener con nuestro monitor de fósforo en el modo 80 columnas han sido sólo tres, frente a las cinco o seis que se pueden obtener en el modo 40 columnas.

Vídeo compuesto en RGBI

El montaje en sí es muy sencillo. No se necesitan conocimientos de electrónica y ni siquiera necesitamos componentes o placas de circuito impreso; en realidad, la aplicación que os proponemos sólo es un conmutador y unos conectores.

La utilidad de este montaje viene dada por la inexistencia del vídeo de 80 columnas en el conector de vídeo compuesto y por otra parte, la ausencia en el conector de vídeo RGBI del vídeo de 40 columnas. Así pues, no tenemos más remedio que proveernos de algo que nos permita introducir el vídeo del conector adecuado a nuestro monitor según deseemos 40 u 80 columnas.

Ante la necesidad de desvelar el misterio de las 80 columnas del 128 y el problema que se nos planteaba a los que ya teníamos un monitor de vídeo compuesto, decidimos estudiar las salidas del conector RGBI con un osciloscopio. Buscábamos una salida que nos proporcionara el vídeo con los sincronismos necesarios para que la imagen fuera estable en la pantalla, así como una salida conectada a masa.

Tuvimos suerte: existían dichas patillas, ¡¡encontramos el vídeo compuesto en el conector del RGBI!!

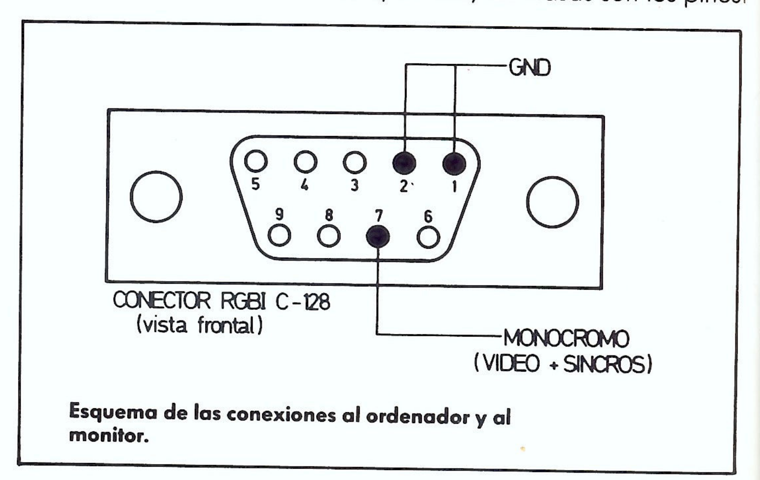
Primero vamos a aclarar un problema que tuvimos con un gráfico del manual, en el apéndice C (titulado conectores/ports para equipos periféricos). A la vista del dibujo del conector de vídeo para RGBI, si nos imaginamos que estamos mirando el conector del ordenador de frente, los números 4 y 5 son las masas y el 8 es el monocromo que nos proporcionará el vídeo con los sincros de 80 columnas. Así como está indicado en la tabla del manual, es como se vería dicho conector desde dentro del ordenador, o también el conector macho que queremos insertar visto de frente. Indicamos esto ya que en un principio esta figura nos indujo a error.

Para evitar problemas, lo mejor es que nos fijemos en la figura que adjuntamos en el artículo; en ella, las masas son los pines.

COMPONENTES

CANTIDAD	DENOMINACION
1	Conmutador 2 cir./2 pos.
1	Conector 9 patillas con caperuza.
1	Conector RCA macho.
1	Conector RCA hembra.
2 metros	Cable coaxial, un conductor y malla.
1	Caja de plástico 8x4x2 cm.

Precio aproximado: 1.100 ptas.



Esquema de las conexiones al ordenador y al monitor.

2 y 1 y el monocromo (vídeo + sincros) es la patilla número 7 (pensemos que estamos mirando el conector del ordenador de frente). Esperamos que de esta forma sea fácil identificarlos rápidamente.

Realización del montaje

Lo primero es lo primero: la lista de materiales necesarios; todos son fáciles de localizar en una tienda de componentes electrónicos.

Las herramientas necesarias son:

a) Un soldador de 30 watios o inferior.
b) Estaño de soldar con alma de resina antioxidante.
c) Un destornillador pequeño.
d) Un instrumento para cortar y pelar el cable.
e) Un instrumento para taladrar la caja que contendrá el circuito de conmutación.

Cuando tengamos todo, debemos preparar la caja. Nosotros hemos utilizado una de plástico con tapa a presión: sus dimensiones son 8 cm de larga, 4 cm de ancha y 2 cm de alta, se deben colocar: el conmutador de dos circuitos/ dos posiciones, el conector RCA hembra y dos taladros para que entren y salgan los cables.

Comenzamos con las soldaduras. Primero hay que identificar en el conector macho de 9 patillas del RGBI las patillas que nos interesan. Hecho esto hemos de soldar el cable apantallado a dichos conectores, uniendo la pantalla del cable a la masa (patillas 4 o 5 del diagrama, es indiferente el que utilicemos) y el vivo a la salida de monocromo (patilla 8 del diagrama).

Se debe tener especial cuidado al efectuar las soldaduras para no establecer cortocircuitos ni puentes entre las distintas patillas del conector. Este cable entrará por uno de los orificios realizados en la caja y lo soldaremos a uno de los circuitos del conmutador, es interesante hacer un nudo al cable por el interior de la caja para evitar que se suelten los contactos ante un eventual tirón.

Posteriormente soldaremos otro cable apantallado entre los terminales del conector RCA hembra que habíamos colocado en la caja, con la pantalla a la masa del conector y el vivo a la parte central del conector, la otra parte irá al otro circuito del conmutador. Debemos tener cuidado de no invertir las conexiones del vivo y la masa en el conmutador con respecto a las que soldamos antes del otro circuito.

Finalmente soldaremos otro cable apantallado al conector RCA macho y el restante extremo del cable, tras pasarlo por el otro orificio de la caja, lo soldaremos a la salida del conmutador. Es conveniente en este caso también, anudar el cable.

Conexiones y puesta en funcionamiento

Primero repasaremos todas las conexiones punto por punto y verificaremos que no hay cortocircuitos. Hecho esto desconectaremos el ordenador y los periféricos de la red; enchufaremos el conector de 9 patillas a la salida RGBI y la salida del cable de vídeo compuesto al conector RCA hembra de la caja, finalmente enchufaremos el conector RCA macho a la entrada de vídeo de nuestro monitor.

Dejando activada la tecla de 80/40 DISPLAY debemos conectar el ordenador y el monitor; si no aparece nada, cambiaremos la posición del conmutador; si aún persiste esta situación, apagaremos todo y verificaremos de nuevo el montaje y conexiones.

Debemos ser pulcros y cuidadosos en el montaje para evitar fallos y posibles daños en el ordenador. Aconsejamos leer cuidadosamente el artículo e intentar comprender todo antes de realizar el montaje, de este modo evitaremos problemas.

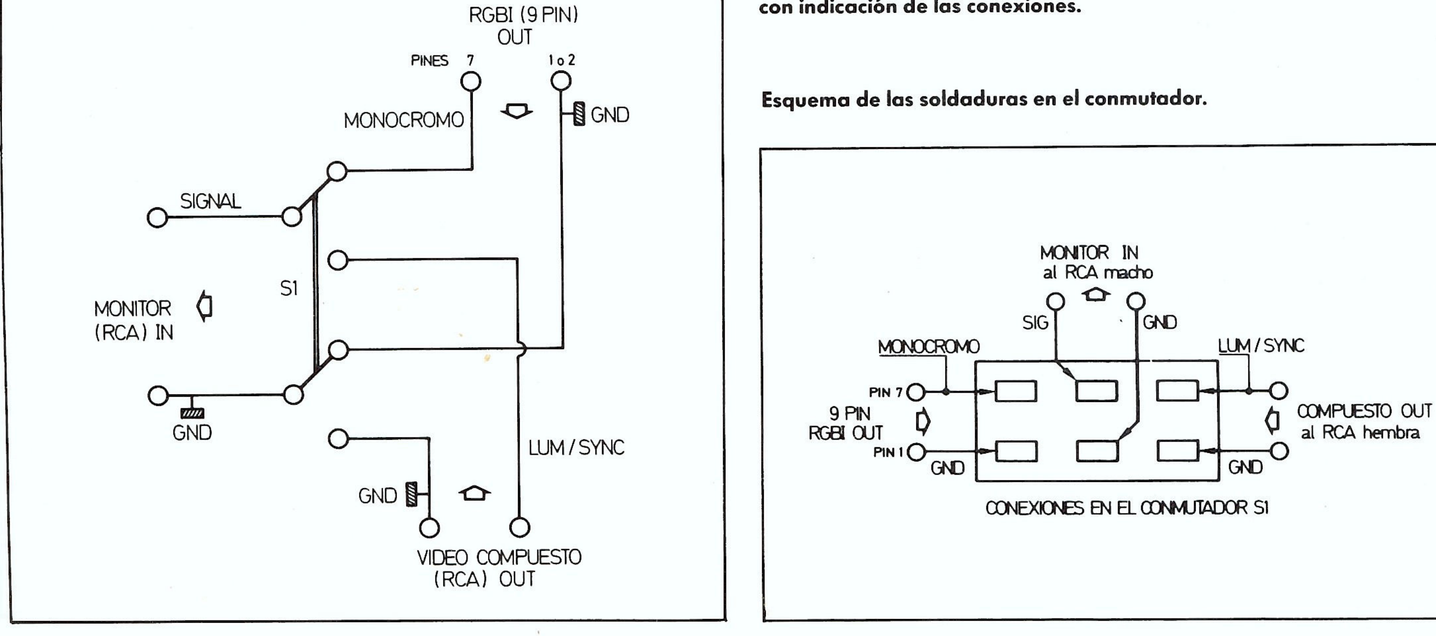
Para mejorar el vídeo de 40 columnas

Aunque nos salgamos un poco del tema de este artículo, creemos interesante que sepáis cómo mejorar la visualización en el modo 40 columnas, tanto en el C-64 como en el C-128 como tal, o emulando al C-64.

Si tenemos un monitor monocromo, podemos mejorar la visualización en 40 columnas, para lo cual debemos tomar la señal de vídeo de la patilla número 1 del conector para vídeo compuesto. En esta patilla tenemos luminancia y sincros de forma que tendremos gradaciones de color pero no nos molestarán las señales de croma (color) presentes en la patilla número 4 que es vídeo out.

De este modo lograremos una mejor definición de los caracteres en los monitores monocromáticos; este truco no vale con los monitores de color de entrada única (vídeo compuesto) ya que perdemos las señales de croma presentes en la patilla número 4 convenientemente mezcladas con el vídeo.

Por otra parte, si poseemos un monitor de color y éste tiene entradas independientes de croma (color out) y de luminancia más sincros, no hay inconveniente para que tomemos el vídeo de la patilla número 1 en lugar de la patilla número 4; en este caso también ganamos resolución.

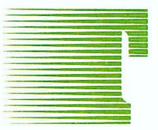


Esquema del circuito y conector RGBI del C-128 con indicación de las conexiones.

Esquema de las soldaduras en el conmutador.

Tiro al plato

Practiquemos el tiro al plato en nuestra propia casa ¡y sin destrozos ni peligro alguno! Tal es la especialidad del programa que a continuación comentamos.

omemos nuestra escopeta de caza, trasladémonos al campo de tiro que refleja la pantalla del televisor, y a la voz de ¡plato! vayamos apuntando uno a uno, o de dos en dos, a los blancos volantes que irán cruzando nuestro campo de tiro.

El juego consiste en hacer blanco en el mayor número de platos que nos sea posible. En esta ocasión, el punto de mira aparecerá en la pantalla, apuntemos con tino —pulsando Z o X, para fijar el objetivo a la izquierda o derecha de la pantalla, respectivamente— y ¡fuego! o más bien ¡barra espaciadora!

La sencillez del programa no deja lugar a dudas: disparar y disparar hasta conseguir sobrepasar los tres niveles de dificultad, y de esta forma alzarnos con el gran trofeo.

En todo caso, por si las complejísimas instrucciones se nos olvidan durante la carga del programa, no hay problema: el menú inicial nos da opción a la entrada a una pantalla de explicación del funcionamiento del juego.

Por otra parte, la segunda opción del menú permite comenzar el juego, para lo cual se efectúa una operación previa de modificación de los caracteres, con la implementación de los gráficos necesarios para la construcción del paisaje en el que se efectúa la competición.

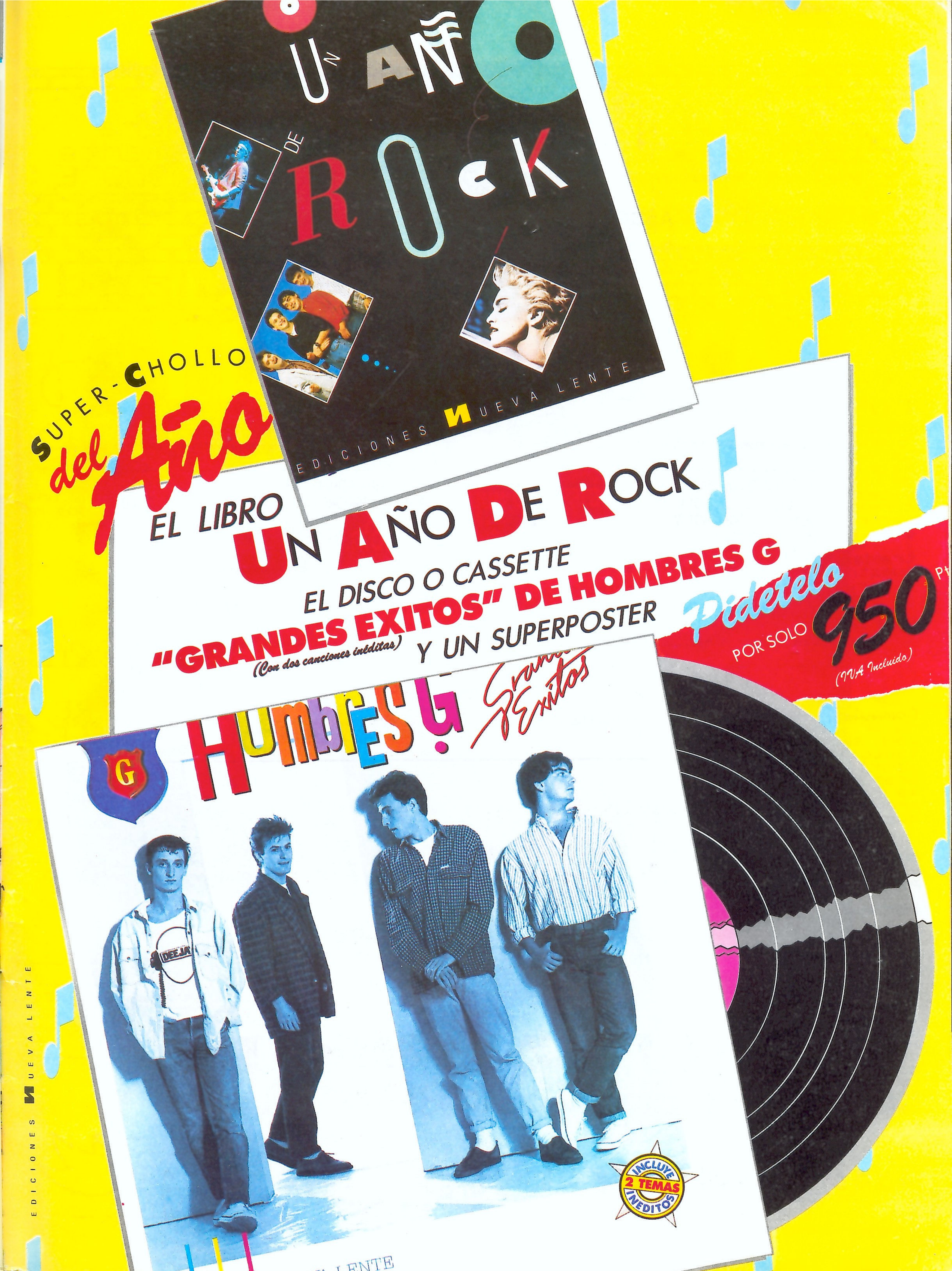
LISTADO

```
0 POKE53280,0:POKE53281,0:PRINT"{CLR}":NI=1:VR=1:PL=1
7:GOTO100                                        -081-
1 POKE56,100:POKE55,0:POKE52,100:POKE51,0       -143-
5 PRINT"{12 ABJ}{6 DCH}ESPERA POR FAVOR..":GOTO  2000
                                                 -224-
8 GOTO 270                                       -074-
10 PRINT"{CLR}{ABJ}{NRJ}{DCH}{RON} |C|<R>|C| {ROF} {R
ON} |C|<R>|C| {ROF} {RON} |3 C| {ROF} {RON} |3 C| {RO
F}[4 ESP]{RON} |3 C| {ROF} {RON} {ROF}"          -023-
20 PRINT"{3 DCH}{RON}|B|{ROF}[5 ESP]{RON}|B|{ROF}[3 E
SP]{RON}|B|{ROF}[3 ESP]{RON}|B|{ROF} {RON}|B|{ROF}[3
ESP]{RON}|B|{ROF}[4 ESP]{RON}|B|{ROF}[3 ESP]{RON}|B|{
ROF} {RON}|B|{ROF}"                              -032-
30 PRINT"{3 DCH}{RON}|B|{ROF}[5 ESP]{RON}|B|{ROF}[3 E
SP]{RON}<Q>|3 C| {ROF} {RON}|B|{ROF}[3 ESP]{RON}|B|{R
OF}[4 ESP]{RON}<Q>|3 C|<W>{ROF} {RON}|B|{ROF}"
                                                 -204-
40 PRINT"{3 DCH}{RON}|B|{ROF}[5 ESP]{RON}|B|{ROF}[3 E
SP]{RON}|B|{ROF} {RON} {ROF}[3 ESP]{RON}|B|{ROF}[3 ES
P]{RON}|B|{ROF}[4 ESP]{RON}|B|{ROF}[3 ESP]{RON}|B|{RO
F} {RON}|B|{ROF}"                                -128-
50 PRINT"{3 DCH}{RON} {ROF}[3 ESP]{RON} |C|<E>|C| {RO
F} {RON} {ROF}[2 ESP]{RON} {ROF}[2 ESP]{RON} |3 C| {R
OF}[4 ESP]{RON} {ROF}[3 ESP]{RON} {ROF} {RON} |3 C| {
ROF}"                                            -246-
60 PRINT"{2 ABJ}{6 DCH}{RON} |3 C| {ROF} {RON} {ROF}[
5 ESP]{RON} |3 C| {ROF} {RON} |C|<R>|C| {ROF} {RON} |
3 C| {ROF}"                                      -142-
70 PRINT"{6 DCH}{RON}|B|{ROF}[3 ESP]{RON}|B|{ROF} {RO
N}|B|{ROF}[5 ESP]{RON}|B|{ROF}[3 ESP]{RON}|B|{ROF}[3
ESP]{RON}|B|{ROF}[3 ESP]{RON}|B|{ROF}[3 ESP]{RON}|B|{
ROF}"                                            -214-
80 PRINT"{6 DCH}{RON}<Q>|3 C| {ROF} {RON}|B|{ROF}[5 E
SP]{RON}<Q>|3 C|<W>{ROF}[3 ESP]{RON}|B|{ROF}[3 ESP]{R
ON}|B|{ROF}[3 ESP]{RON}|B|{ROF}"                 -130-
90 PRINT"{6 DCH}{RON}|B|{ROF}[5 ESP]{RON}|B|{ROF}[5 E
SP]{RON}|B|{ROF}[3 ESP]{RON}|B|{ROF}[3 ESP]{RON}|B|{R
OF}[3 ESP]{RON}|B|{ROF}[3 ESP]{RON}|B|{ROF}"
                                                 -146-
95 PRINT"{6 DCH}{RON} {ROF}[5 ESP]{RON} |3 C| {ROF} {
RON} {ROF}[3 ESP]{RON} {ROF}[3 ESP]{RON} {ROF}[3 ESP]
{RON} |3 C| {ROF}"                               -087-
97 GOTO 130                                      -125-
100 CIV=53248:POKECIV+23,1:POKECIV+29,1          -076-
110 POKECIV+21,1:POKE2040,195                    -168-
120 FORT=0TO62:READA:POKE195*64+T,A:NEXTT:POKECIV+39,
12:GOTO 10                                       -183-
130 POKECIV+21,1:POKECIV,155:POKECIV+1,165       -049-
150 PRINT"{YEL}{HOM}{21 ABJ}{2 DCH}{RON}|\|[4 ESP]<*>
";:PRINT"{16 DCH}{RON}|\|[4 ESP]<*>{ROF}"        -179-
155 PRINT"{2 DCH}{RON}[2 ESP]F1[2 ESP]{ROF}{WHT} INTR
UCCIONES{YEL}";:PRINT"{3 DCH}{RON}[2 ESP]F3[2 ESP]{RO
F}{WHT} TO LOAD{YEL}"                            -154-
160 PRINT"{2 DCH}<*>{RON}[4 ESP]{ROF}|\|";:PRINT"{16
DCH}<*>{RON}[4 ESP]{ROF}|\|"                     -048-
170 GETK$:IF K$="" THEN 170                      -231-
180 IF K$<>"{F1 }" AND K$<>"{F3 }" THEN 170      -231-
190 IF K$="{F1 }" THEN 7500                      -025-
200 PRINT"{CLR}":POKECIV+21,0:GOTO 1             -015-
250 :                                            -241-
260 :                                            -242-
270 GOSUB 700                                    -240-
275 POKECIV+21,255:POKECIV+21,0                  -090-
280 POKE19,1:INPUT"{CLR}{WHT}{11 ABJ}JUGADOR -1- MAX.
8 LETRAS..";N$(1)                                -170-
290 IF LEN(N$(1))>8 THEN 280                     -013-
300 INPUT"{CLR}{11 ABJ}JUGADOR -2- MAX.8 LETRAS..";N$
(2)                                              -112-
310 IF LEN(N$(2))>8 THEN 300                     -000-
320 POKE19,0                                     -170-
325 :                                            -244-
330 :                                            -240-
335 :                                            -245-
350 PRINT"{CLR}"                                 -028-
400 PRINT"{BLK}{HOM}{ARB}<30 U>"                 -073-
410 PRINT"<30 U>"                                -022-
420 FORT=1TO21                                   -101-
430 PRINT"{BLK}{29 DCH}<U>":NEXTT                -230-
450 PRINT"<30 U>"                                -026-
460 YU=56256:POKEYU,0:FORT=1984 TO 2013          -081-
470 POKET,120:POKEYU+1,0:YU=YU+1:NEXT T          -232-
482 PRINT"{HOM}{ARB}{23 ABJ}<30 U>"              -074-
490 FORT=1TO2000:NEXTT:POKE53280,0:POKE53281,14
                                                 -092-
500 PRINT"{HOM}{ARB}{7 ABJ}{18 DCH}{YEL}<YEL><F>"
                                                 -190-
510 PRINT"{4 DCH}{MRN}|A||B||C||D|{7 DCH}|Y||Z||+|{YE
L}<C><X>{3 DCH}{MRN}<S><P><A>"                   -003-
520 PRINT"{3 DCH}|*|<3 U>|E||F|{2 DCH}|N||Q|{DCH}|W||
X|<U>|U|{DCH}<*><G>{DCH}<Z><2 U><E><R>"          -201-
530 PRINT"{2 DCH}>?<4 U>|G|{DCH}|M|<U>|R|{DCH}|V|<3 U
>|H|<K><U>|\|<D><3 U><W>"                        -199-
540 PRINT"{2 DCH}=<6 U>|H||L|<2 U>|U|<4 U>{PI }<I><U>
<N><5 U><H>"                                     -089-
550 PRINT"{DCH}<<7 U>|I||K|<U>|S|<O><5 U><T><+><Q><5
U><J>"                                           -157-
560 PRINT"{DCH};<8 U>|J||O||T|<6 U><@><M><7 U><L>"
                                                 -106-
570 PRINT":<10 U>|P|<8 U><\><7 U><Y>"            -238-
580 PRINT"<29 U>"                                -102-
590 PRINT"{BLU}<29 U>"                           -134-
595 FORT=1TO6:PRINT"{GRN}<29 U>":NEXTT           -019-
698 GOSUB 2500;GOTO 800                          -089-
700 FORT=0TO62:READZA:POKE5*64+T,ZA:NEXTT        -160-
710 FORT=0TO62:READZA:POKE14*64+T,ZA:NEXTT       -209-
720 FORT=0TO62:READZA:POKE15*64+T,ZA:NEXTT       -211-
730 FORT=0TO62:READZA:POKE225*64+T,ZA:NEXTT      -007-
740 FORT=0TO62:READZA:POKE250*64+T,ZA:NEXTT      -006-
760 RETURN                                       -157-
800 GOSUB 3000:GOSUB 2500:FORT=54272 TO 54296:POKET,0
:NEXTT                                           -069-
810 POKE54296,15:POKE54277,25:POKE54278,40       -140-
820 POKE54272,126:POKE54273,38                   -183-
900 CIV=53248:POKECIV+41,8:POKECIV+40,2:POKECIV+28,28
                                                 -203-
905 POKECIV+23,31:POKECIV+29,31                  -096-
910 POKE 2040,5:POKE2041,14:POKE2042,15:POKE2043,225:
POKE2044,250                                     -161-
915 POKECIV+37,0:POKECIV+38,2:POKECIV+39,2:POKECIV+21
,7                                               -124-
929 POKECIV+42,1:POKECIV+43,1                    -253-
930 POKECIV,76:POKECIV+1,65                      -164-
940 POKECIV+2,157:POKECIV+3,65                   -052-
942 POKECIV+4,125:POKECIV+5,192                  -102-
945 POKECIV+6,125:POKECIV+7,192                  -109-
947 POKECIV+8,125:POKECIV+9,192                  -115-
990 :                                            -252-
1000 PRINT"{HOM}":CA=1185:CE=1185:ZA=1211:ZE=1211:M=0
:TI$="[6 0]"                                     -159-
1010 X=INT(RND(0)*10)+1:A=INT(RND(0)*10)+1       -063-
1015 PRINT"{HOM}{ARB}{9 ABJ}{6 IZQ}";RIGHT$(TI$,2)
                                                 -240-
1020 IF X=1 THEN C=1:GOTO 1040                   -208-
1030 D=1                                         -150-
1040 IF A=1 THEN E=1:GOTO 1060                   -191-
1050 F=1                                         -154-
1060 IF C=1 THEN CA=CA+1:POKECA,126:POKECA-1,32
                                                 -229-
1070 IF D=1 THEN CE=CE+1:POKECE,126:POKECE-1,32
                                                 -247-
1080 IF CA=1196 THENPOKECA,32:CA=1185:C=0        -113-
1090 IF CE=1196 THENPOKECE,32:CE=1185:D=0        -127-
1100 IF E=1 THEN ZA=ZA-1:POKEZA,126:POKEZA+1,32
                                                 -062-
```

```
1110 IF F=1 THEN ZE=ZE-1:POKEZE,126:POKEZE+1,32
                                                       -080-
1120 GETK$:IF K$="" THEN 1160                          -067-
1130 IF K$="Z" THEN POKECIV+21,19:POKECIV+39,2:POKECI
V+40,12:JA=1:JE=0                                      -148-
1140 IF JA=1 OR JE=1 THEN IFK$=" "THEN DIS=1:GOSUB 20
320                                                    -232-
1150 IF K$="X" THEN POKECIV+21,11:POKECIV+39,12:POKEC
IV+40,2:JE=1:JA=0                                      -140-
1160 IFZA=1200THENPOKEZA,32:ZA=1211:E=0                -095-
1170 IFZE=1200THENPOKEZE,32:ZE=1211:F=0                -109-
1180 IFJA=1ANDDIS=1THENIFCA=1193ORCE=1193THENW(RU)=W(
RU)+1                                                  -122-
1190 IFJE=1ANDDIS=1THENIFZA=1203ORZE=1203THENW(RU)=W(
RU)+1                                                  -157-
1195 PRINT"{HOM}{ARB}{14 ABJ}{7 IZQ}";W(RU)            -040-
1200 DIS=0                                             -048-
1210 IF TI$="[4 0]30"THEN PRINT"{HOM}{ARB}{9 ABJ}{6 I
ZQ}30":GOSUB4000:GOSUB2500:GOTO800                     -112-
1220 GOTO 1010                                         -000-
2000 POKE52,48:POKE56,48                               -124-
2010 POKE56334,PEEK(56334)AND254                       -044-
2020 POKE1,PEEK(1)AND251                               -130-
2030 FORI=0TO511:POKEI+12288,PEEK(I+53248):NEXT
                                                       -051-
2040 POKE1,PEEK(1)OR4                                  -238-
2050 POKE56334,PEEK(56334)OR1                          -148-
2060 POKE53272,(PEEK(53272)AND240)+12                  -007-
2070 FORI=12752 TO 13055:READC:POKEI,C:NEXT I
                                                       -212-
2080 FORI=13064 TO 13303:READA:POKEI,A:NEXT I
                                                       -202-
2090 FORI=12296 TO 12503:READK:POKEI,K:NEXT I
                                                       -230-
2100 FORI=12672 TO 12751:READW:POKEI,W:NEXT I:GOTO 8
                                                       -196-
2500 PRINT"{HOM}{4 ABJ}{9 IZQ}[8 ESP]"                 -148-
2505 IF KL=0 THEN JU=1                                 -027-
2507 PRINT"{BLK}{HOM}{2 ABJ}{9 IZQ}JUGADOR"            -021-
2510 PRINT"{2 ABJ}{7 IZQ}"; JU                         -032-
2515 IF KL=0 THEN JU=0                                 -027-
2520 PRINT"{3 ABJ}{8 IZQ}TIEMPO"                       -163-
2530 PRINT"{WHT}{2 ABJ}{6 IZQ}00"                      -240-
2540 PRINT"{BLK}{3 ABJ}{8 IZQ}PLATOS"                  -058-
2550 PRINT"{2 ABJ}{6 IZQ}[2 ESP]"                      -205-
2560 PRINT"{3 ABJ}{9 IZQ}{WHT}NIVEL";NI                -203-
2570 PRINT"{3 ABJ}{8 IZQ}AUTOR"                        -101-
2580 PRINT"{2 ABJ}{9 IZQ}FRANCESC"                     -172-
2590 PRINT"{2 ABJ}{9 IZQ}BARDERA"                      -089-
2595 PRINT"{HOM}{ARB}{7 DCH}{WHT}MINIMO";PL;"{IZQ} PL
ATOS"                                                  -233-
2600 RETURN                                            -200-
3000 IF GE=0THENIF JU=2 THEN JU=0:RU=0                 -064-
3003 IF HA=0 THENIFRU=2 THEN RU=Q                      -002-
3005 KL=1                                              -237-
3007 IFHA=0 THENJU=JU+1:RU=RU+1                        -182-
3010 PRINT"{BLK}{HOM}{ARB}{5 ABJ}{4 DCH}PREPARADO JUG
ADOR{YEL}";JU                                          -244-
3050 FORT=1TO 2500:NEXT T:PRINT"{HOM}{ARB}{5 ABJ}{3 D
CH}[24 ESP]"                                           -099-
3100 RETURN                                            -196-
4000 KL=1:GOSUB 7000                                   -138-
4010 IFJU=1THENIFW(RU)<PLTHENPRINT"{HOM}{ARB}{5 ABJ}{
4 DCH} PRUEBA NO SUPERADA":W=0:GU=1:RU=2               -118-
4012 IFJU=1THENIFW(RU)<PLTHEN JU=2:HA=1:GE=1:Y=Y+1:GO
TO 4027                                                -015-
4015 IFJU=2THENIFW(RU)<PLTHENPRINT"{HOM}{ARB}{5 ABJ}{
4 DCH} PRUEBA NO SUPERADA":GOTO 4500                   -070-
4020 PRINT"{HOM}{ARB}{5 ABJ}{5 DCH}PRUEBA SUPERADA !!
"                                                      -215-
4027 IF KO=0 THEN W(1)=0:W(2)=0                        -217-
4029 FOR H=1 TO 2500:NEXT H                            -055-
4030 PRINT"{HOM}{ARB}{5 ABJ}[28 ESP]"                  -049-
4032 IF Y=2 THEN Y=0:GOTO 5500                         -242-
4035 IF Y=0 THEN VR=VR+1                               -185-
4037 IF GU=1 THEN VR=VR+1:GU=0                         -066-
4040 IF VR=2 THEN VR=0:GOTO 4150                       -143-
4050 IF NI=2 THEN 4070                                 -216-
4060 IF NI=1 THEN NI=NI+1:PL=20:W(1)=0:W(2)=0:GOTO 41
50                                                     -167-
4070 IF NI=3 THEN 5000                                 -213-
4080 NI=NI+1:PL=23:KO=1:W(1)=0:W(2)=0                  -110-
4150 RETURN                                            -202-
4500 W=0:RU=1:JU=1:HA=1:GE=1:Y=Y+1:GOTO 4027
                                                       -141-
5000 IF W(1)>W(2) THEN RU=1                            -009-
5010 IF W(1)=W(2)THEN GOSUB 6000:PRINT"{HOM}{ARB}{3 A
BJ}{5 DCH}ENPATE!! PRUEBA OTRA":GOTO                   -084-
5020 IF W(2)>W(1) THEN RU=2                            -012-
5050 GOSUB 7000                                        -081-
5060 PRINT"{BLK}{HOM}{ARB}{3 ABJ}{7 DCH}GANADOR {WHT}
";N$(RU):GOTO 6000                                     -013-
5500 PRINT"{WHT}{HOM}{ARB}{3 ABJ}{9 DCH}T H E[3 ESP]E
 N D":GOTO 6000                                        -141-
6000 PRINT"{BLK}{HOM}{ARB}{5 ABJ}{3 DCH}QUIERES JUGAR
 OTRA (S/N)"                                           -229-
6010 GETK$:IFK$=""THEN6010                             -229-
6020 IF K$<>"S"ANDK$<>"N"THEN6010                      -091-
6030 IFK$="N"THENPRINT"{CLR}":POKECIV+21,0:END
                                                       -143-
6035 VR=1:RU=0:GE=0                                    -133-
6040 POKE53280,0:POKE53281,0:KL=0:NI=1:PL=17:W(1)=0:W
(2)=0:HA=0:GU=0:GOTO280                                -184-
7000 PRINT"{HOM}{ARB}{4 ABJ}[29 ESP]":POKECIV+21,0
                                                       -117-
7010 RETURN                                            -200-
7500 POKECIV+21,0:PRINT"{CLR}{HOM}{6 ABJ}{3 DCH}SUPER
A LOS 3 NIVELES PARA GANAR"                            -137-
7510 PRINT"{ABJ}{3 DCH}CADA NIVEL TIENE UN MINIMO DE
PLATOS"                                                -170-
7520 PRINT"{ABJ}{3 DCH}LAS TECLAS SON "                -019-
7530 PRINT"{2 ABJ}{8 DCH}'Z' IZQUIERDA 'X' DERECHA"
                                                       -018-
7540 PRINT"{ABJ}{10 DCH}[3 ESP]' ' DISPARO"            -244-
7550 PRINT"{4 ABJ}{14 DCH}{RON}PULSA TECLA{ROF}"
                                                       -078-
7560 GETT$:IFT$="" THEN7560                            -045-
7570 GOTO 10                                           -173-
20000 DATA0,0,0,0,0,0,0,0,0,0,0,0,4,0,32,2,0,64,1,0,1
28,0,129,0,0,66,0,0,36,0                               -138-
20010 DATA0,24,0,0,24,0,0,36,0,16,66,8,9,129,144,7,0,
224,14,0,112,29,0,184                                  -079-
20020 DATA56,129,28,48,0,12,0,0,0                      -142-
20025 REM ***** CARACTERES *****                       -254-
20030 DATA1,3,7,15,31,63,127,255,15,31,31,63,127,127,
255,255,0,1,1,3,3,7,7,15                               -246-
20040 DATA7,15,31,31,63,63,127,127,0,0,0,0,1,3,3,7,63
,127,255,255,255,255,255                               -000-
20045 DATA255,0,1,3,3,7,15,31,63,0,1,3,15,31,63,127,1
27,31,255,255,255,255,255                              -053-
20050 DATA255,255,128,192,240,248,255,255,255,255,0,0
,0,0,0,192,240,248                                     -240-
20055 DATA252,254,255,255,255,255,255,255,0,0,0,0,128
,128,192,224,240,240,248                               -030-
20060 DATA252,254,254,255,255,128,192,224,224,224,240
,253,252,253,253,252,254                               -041-
20065 DATA255,255,255,255,223,207,239,231,243,249,252
,254,255,255,255,127,127                               -063-
20070 DATA63,191,159,63,127,127,255,255,255,255,255,1
,3,3,7,15,31,31,63,0,0,0                               -016-
20075 DATA28,63,127,255,255,255,255,255,255,255,255,2
55,126,62,158,220,201                                  -167-
20080 DATA227,243,249,252,0,0,0,0,0,128,192,224,240,2
40,248,252,252,254,254                                 -110-
20090 DATA255,254,254,252,249,251,243,247,231,207,223
,223,159,63,127,127                                    -058-
20095 DATA1,129,195,199,239,207,159,191,7,15,15,31,63
,127,127,255,0,0,0,0,0                                 -186-
20100 DATA1,3,3,15,31,63,127,255,255,255,255,0,0,0,0,
0,1,7,15,0,0,0,0,126                                   -012-
20110 DATA255,255,255,0,0,0,0,0,128,128,192,224,240,2
48,252,254,254,255,255                                 -175-
20115 DATA128,192,224,224,240,240,248,248,252,252,254
,254,255,255,255,255                                   -110-
```

```
20120 DATA0,0,0,0,0,1,3,3,7,7,7,15,15,31,31,63,63,63,
127,127,127,63,191,191                             -139-
20125 DATA159,223,207,239,239,231,247,247,243,251,249
,253,252,254,254,254                               -119-
20130 DATA0,0,112,248,252,254,254,255,255,255,254,2
54,252,249,251                                     -047-
20135 DATA243,231,239,207,223,63,255,127,127,63,191,1
59,223,255,255,255,0,0                             -197-
20140 DATA128,128,128,192,192,224,225,227,227,231,239
,207,223,159                                       -224-
20145 DATA191,63,127,127,255,255,255,255,31,31,63,63,
127,127,255,255,0,0,1,3,3                          -075-
20150 DATA7,7,15,0,0,0,3,7,15,31,63,0,24,254,255,255,
255,255,255,0,0,0,0,192                            -192-
20155 DATA224,248,252,254,254,255,255,255,255,255,255
,0,0,0,128,192,192,224,224                         -143-
20160 DATA240,248,248,252,252,254,254,255,128,128,192
,192,224,224,240,240                               -103-
20165 DATA248,248,252,254,254,255,255,255,128,192,192
,224,224,240,240,248                               -120-
20170 DATA248,252,252,254,254,255,255,255,255,255,255
,255,255,255,255,255                               -130-
20175 DATA63,127,255,255,255,255,255,255,0,0,3,31,63,
63,127,127,0,0,192,248,252                         -132-
20180 DATA252,254,254,127,127,63,63,31,3,0,0,254,254,
252,252,248,192,0,0                                -027-
20182 DATA24,126,126,255,255,126,126,24            -211-
20183 :                                            -088-
20184 DATA24,36,66,126,66,66,66,0,124,34,34,60,34,34,
124,0,28,34,64,64,64,34,28                         -137-
20185 DATA0,120,36,34,34,34,36,120,0,126,64,64,120,64
,64,126,0,126,64,64,120,64                         -106-
20186 DATA64,64,0,28,34,64,78,66,34,28,0,66,66,66,126
,66,66,66,0,28,8,8,8,8,8                           -073-
20187 DATA28,0,14,4,4,4,4,68,56,0,66,68,72,112,72,68,
66,0,64,64,64,64,64,64,126                         -147-
20188 DATA0,66,102,90,90,66,66,66,0,66,98,82,74,70,66
,66,0,24,36,66,66,66,36,24                         -169-
20189 DATA0,124,66,66,124,64,64,64,0,24,36,66,66,74,3
6,26,0,124,66,66,124,74,68                         -156-
20190 DATA66,0,60,66,64,60,2,66,60,0,62,8,8,8,8,8,8,0
,66,66,66,66,66,66,60                              -158-
20191 DATA0,66,66,66,36,36,24,24,0,66,66,66,90,90,102
,66,0,66,66,36,24,36,66,66                         -154-
20192 DATA0,34,34,34,28,8,8,8,0,126,2,4,24,32,64,126,
0,60,66,70,90,98,66,60,0,8                         -117-
20193 DATA24,40,8,8,8,62,0,60,66,2,12,48,64,126,0,60,
66,2,28,2,66,60,0,4,12,20                          -048-
20194 DATA36,126,4,4,0,126,64,120,4,2,68,56,0,28,32,6
4,124,66,66,60,0,126,66,4                          -068-
20195 DATA8,16,16,16,0,60,66,66,60,66,66,60,0,60,66,6
6,60,2,4,56,0                                      -255-
20196 :                                            -092-
20200 DATA0,0,0,0,0,0,0,0,0,0,0,0,0,0,3,255,192,3,2
55,192,3,0,192,3,0,192                             -074-
20210 DATA3,0,192,3,0,192,3,0,192,3,0,192,3,0,192,3,2
55,192,3,255,192                                   -112-
20220 DATA0,0,0,0,0,0,0,0,0,0,0,0,0,0              -104-
20222 :                                            -082-
20230 DATA0,0,0,0,0,0,0,0,0,0,0,0,0,0,3,255,192,3,2
55,192,3,0,192,3,0,192                             -077-
20240 DATA3,0,192,3,0,192,3,0,192,3,0,192,3,0,192,3,2
55,192,3,255,192                                   -115-
20245 DATA0,0,0,0,0,0,0,0,0,0,0,0,0,0              -111-
20250 :                                            -083-
20260 DATA0,8,0,0,8,0,0,8,0,0,56,0,0,248,0,0,248,0,3,
200,0,1,72,0,5,88,0                                -207-
20270 DATA5,91,0,5,91,0,5,91,192,5,88,192,53,88,240,5
3,88,240,63,255,240                                -054-
20280 DATA63,255,192,63,255,0,63,240,0,63,240,0,63,24
0,0                                                -001-
20281 :                                            -087-
20285 DATA0,0,0,0,32,0,0,32,0,0,160,0,3,128,0,15,12
8,0,254,128,1,114,0                                -204-
20290 DATA5,82,0,5,90,0,5,91,0,5,91,192,5,88,192,53,8
8,240,53,88,240                                    -107-
20295 DATA63,255,240,63,255,192,63,255,0,63,240,0,63,
240,0,63,240,0                                     -038-
20296 :                                            -093-
20300 DATA0,0,8,0,0,8,0,0,10,0,0,2,192,0,2,240,0,2,
191,0,0,141,64,0,133,80                            -122-
20305 DATA0,165,80,0,229,80,3,229,80,3,37,80,15,37,92
,15,37,92,15,255,252                               -091-
20310 DATA3,255,252,0,255,252,0,15,252,0,15,252,0,15,
252                                                -242-
20315 :                                            -085-
20320 POKE54276,129:FORT=1TO5:NEXTT:POKE54276,128:RET
URN                                                -240-

READY.                                             -173-
```


Un Año de Rock
Ediciones Nueva Lente
Super-Chollo del Año
El libro
Un Año de Rock
El disco o cassette
"Grandes Exitos" de Hombres G
(Con dos canciones inéditas)
y un superposter
Pídetelo
por solo 950 Pts
(IVA Incluido)
Hombres G
Grandes Exitos
Incluye 2 temas inéditos
Ediciones Nueva Lente

EL **BOMBAZO** del año : la primera tienda exclusivamente dedicada a COMMODORE :

Compuland

Corre a COMPULAND

Alli podras encontrar todo lo que las demas tiendas del pais no quisieron importar por considerarlo de un nivel superior a las necesidades de un usuario normal: nosotros no hemos cometido este error y por tanto tenemos lo que ningun otro puede ofrecer:

-Las ultimas novedades de SOFTWARE, traidas especialmente por nosotros y para nuestros clientes.

-Lo mas reciente en Hardware.

-Un unico y exclusivo CLUB DE SOFT .

-Y una atencion muy especial al disco y al 128.

NO LLORES

Hemos creado un CLUB DE SOFTWARE para que tengas acceso rapido y muy comodo de cualquier juego disponible en la tienda.

Asi mismo dispondras de la informacion mas relevante mediante una revista mensual ,que tratara de todos los temas de interes para el usuario que realmente este atraido por su micro.

Compuland dispone ya de la mayor seleccion de soft del pais, selecion que incluye tanto programas nacionales como importaciones directas de Gran Bretana y Estados Unidos.Cada mes se ira aumentando la biblioteca de programas disponibles importando todo lo que tenga un interes real para el usuario:

-Programas de diseno asistido por ordenador (tipo CAD).

-Programas musicales que te permitiran hasta la mezcla de voces.

-Programas inter-relacionados que combinan hoja de calculo, base de datos...





ASOMBRATE DEL HARDWARE DISPONIBLE:

-multiplicadores de velocidad de disco

-Digitalizadores

-Samplers

-Tabletas graficas

-Taladradores de disco

-Lapices opticos de alta precision y un largo etc.....

DISCO

COMPULAND te ofrece ya una inmejorable gama de programas elegidos en las listas de numeros unos americanos e ingleses.Es ya una realidad y como tal se mantendra una perfecta correlacion con el soft de calidad (cualquiera que sea su pais de origen), tratando de ofrecer siempre una relacion calidad/precio lo mas ajustada posible:

-juegos

-utilidades

-herramientas

Y una sensacional oferta en DISCOS VIRGENES : LLAMANOS Y TE ASOMBRARAS.

